Fig.1

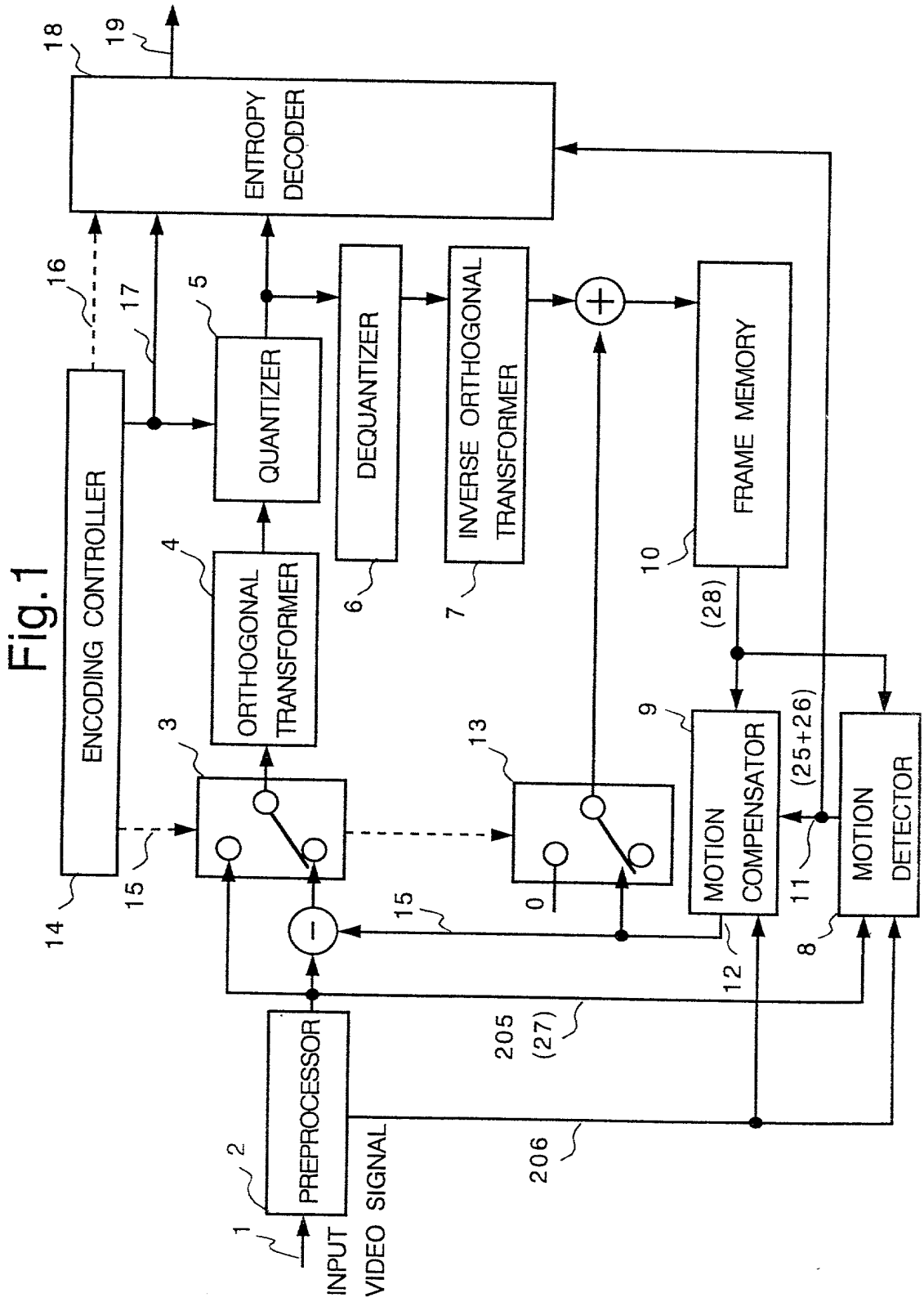

INPUT VIDEO SIGNAL
1
PREPROCESSOR
2
ENCODING CONTROLLER
14
15
16
17
18
19
ENTROPY DECODER
3
ORTHOGONAL TRANSFORMER
4
QUANTIZER
5
DEQUANTIZER
6
INVERSE ORTHOGONAL TRANSFORMER
7
FRAME MEMORY
10
(28)
MOTION COMPENSATOR
9
MOTION DETECTOR
8
11
(25+26)
12
13
0
205
(27)
206

Fig.2

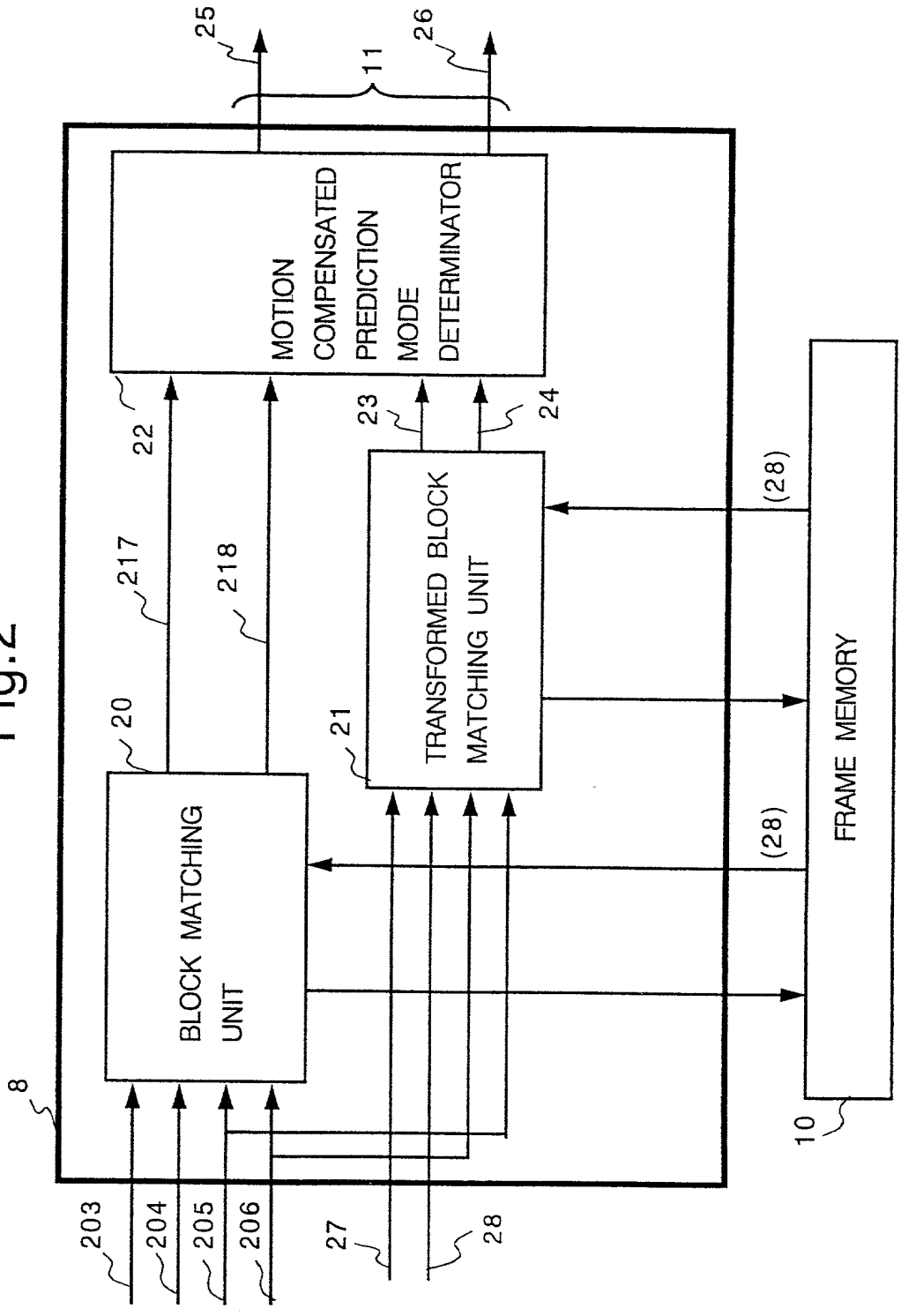
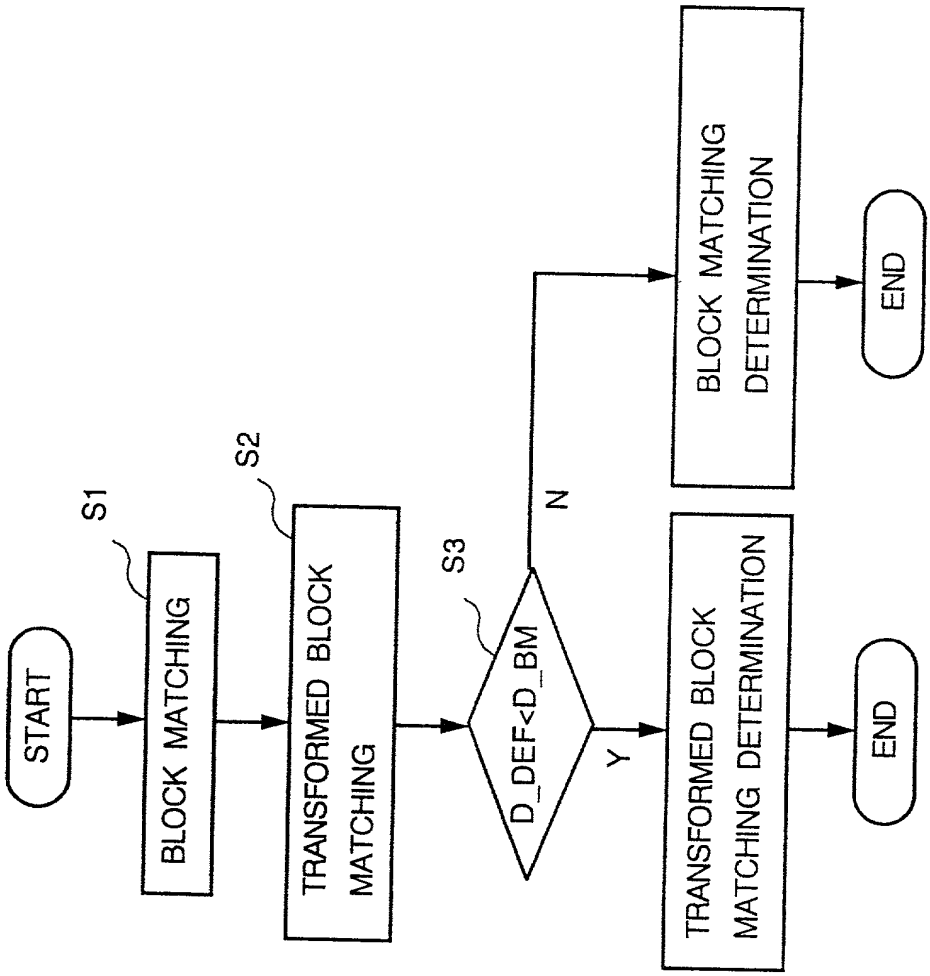

8
BLOCK MATCHING UNIT
20
217
218
MOTION COMPENSATED PREDICTION MODE DETERMINATOR
22
25
11
26
TRANSFORMED BLOCK MATCHING UNIT
21
23
24
(28)
(28)
FRAME MEMORY
10
203
204
205
206
27
28

Fig.3


START
S1
BLOCK MATCHING
S2
TRANSFORMED BLOCK MATCHING
S3
D_DEF<D_BM
Y
N
TRANSFORMED BLOCK MATCHING DETERMINATION
BLOCK MATCHING DETERMINATION
END
END

EPO - DG 1 - 06.06.2006

Fig.4

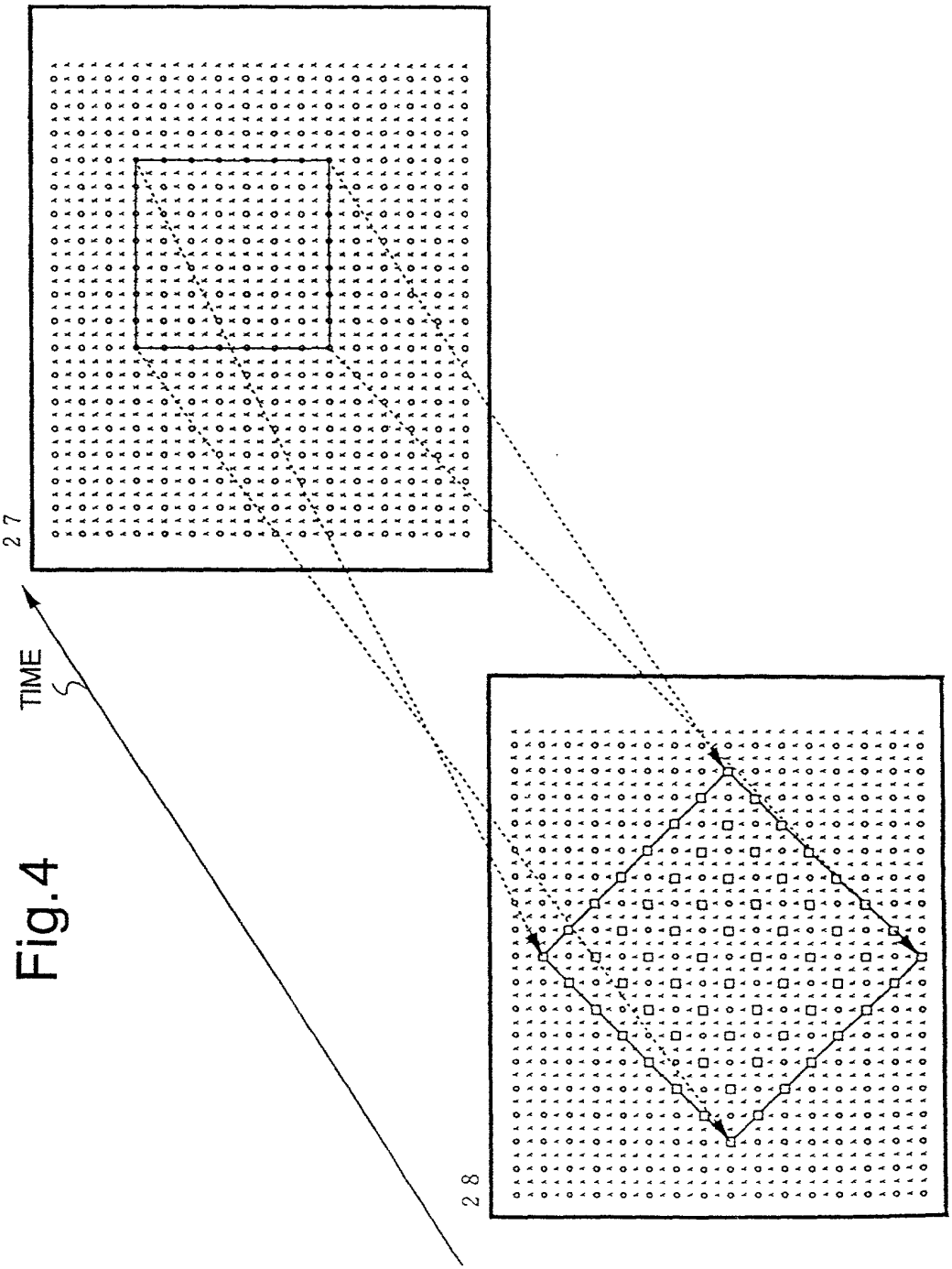


Fig.5

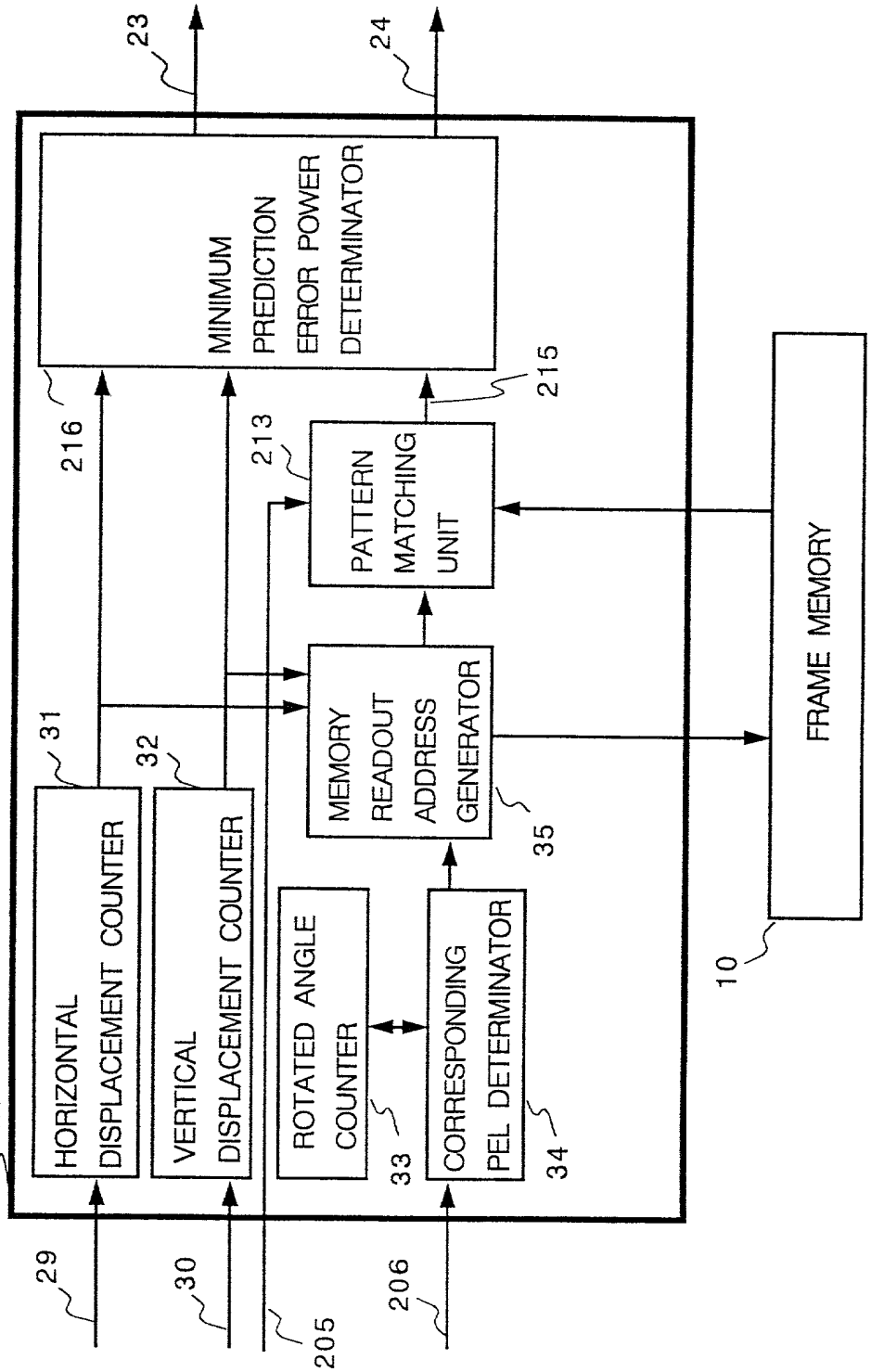

21
HORIZONTAL DISPLACEMENT COUNTER
31
VERTICAL DISPLACEMENT COUNTER
32
ROTATED ANGLE COUNTER
33
CORRESPONDING PEL DETERMINATOR
34
MEMORY READOUT ADDRESS GENERATOR
35
PATTERN MATCHING UNIT
213
MINIMUM PREDICTION ERROR POWER DETERMINATOR
216
215
23
24
29
30
205
206
10
FRAME MEMORY

Fig.6

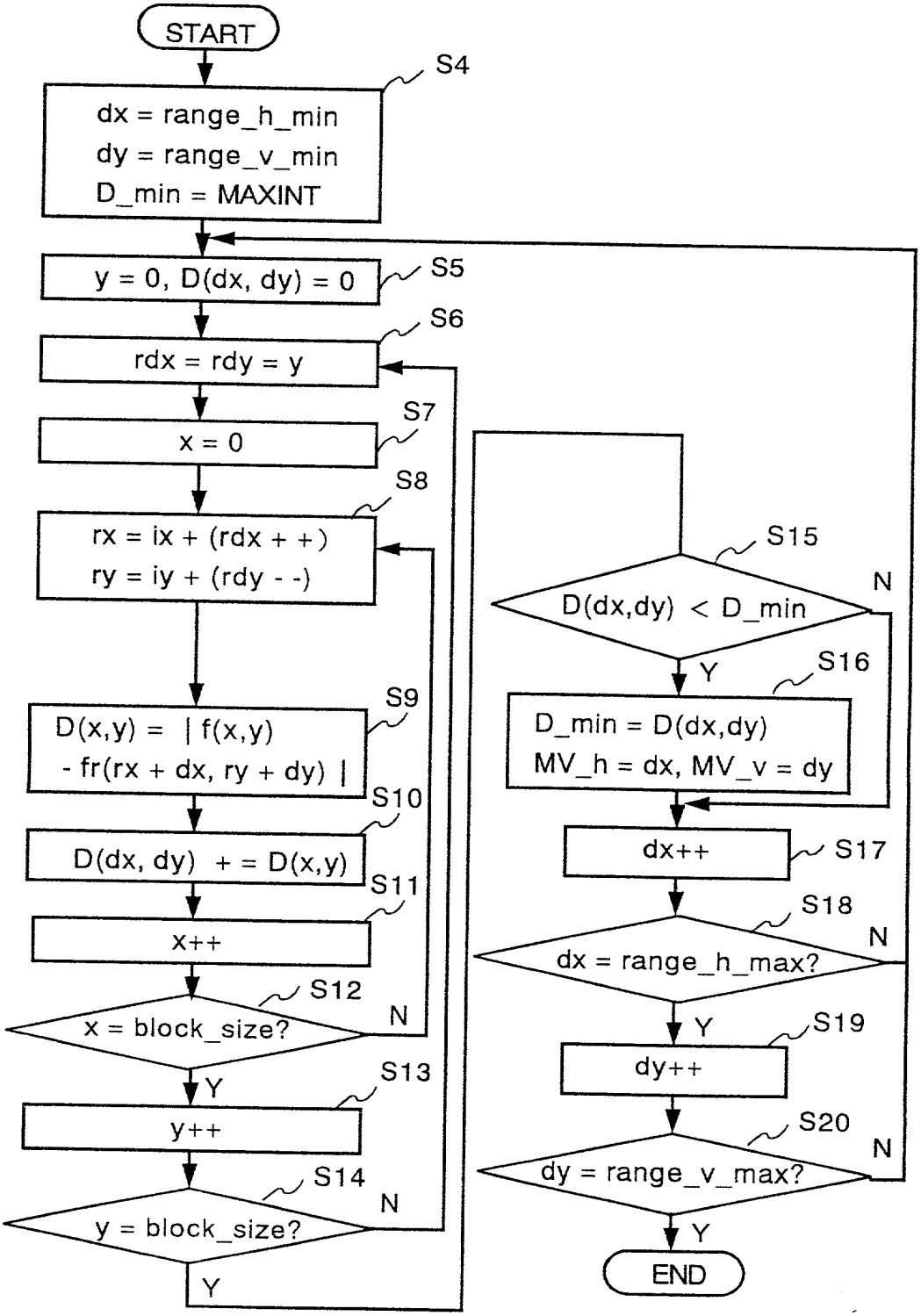

START
S4
dx = range_h_min
dy = range_v_min
D_min = MAXINT
y = 0, D(dx, dy) = 0
S5
S6
rdx = rdy = y
x = 0
S7
S8
rx = ix + (rdx + +)
ry = iy + (rdy - -)
S9
D(x,y) = | f(x,y)
- fr(rx + dx, ry + dy) |
S10
D(dx, dy) + = D(x,y)
S11
x++
S12
x = block_size?
N
Y
S13
y++
S14
y = block_size?
N
Y
S15
D(dx,dy) < D_min
N
Y
S16
D_min = D(dx,dy)
MV_h = dx, MV_v = dy
dx++
S17
S18
dx = range_h_max?
N
Y
S19
dy++
S20
dy = range_v_max?
N
Y
END

Fig.7

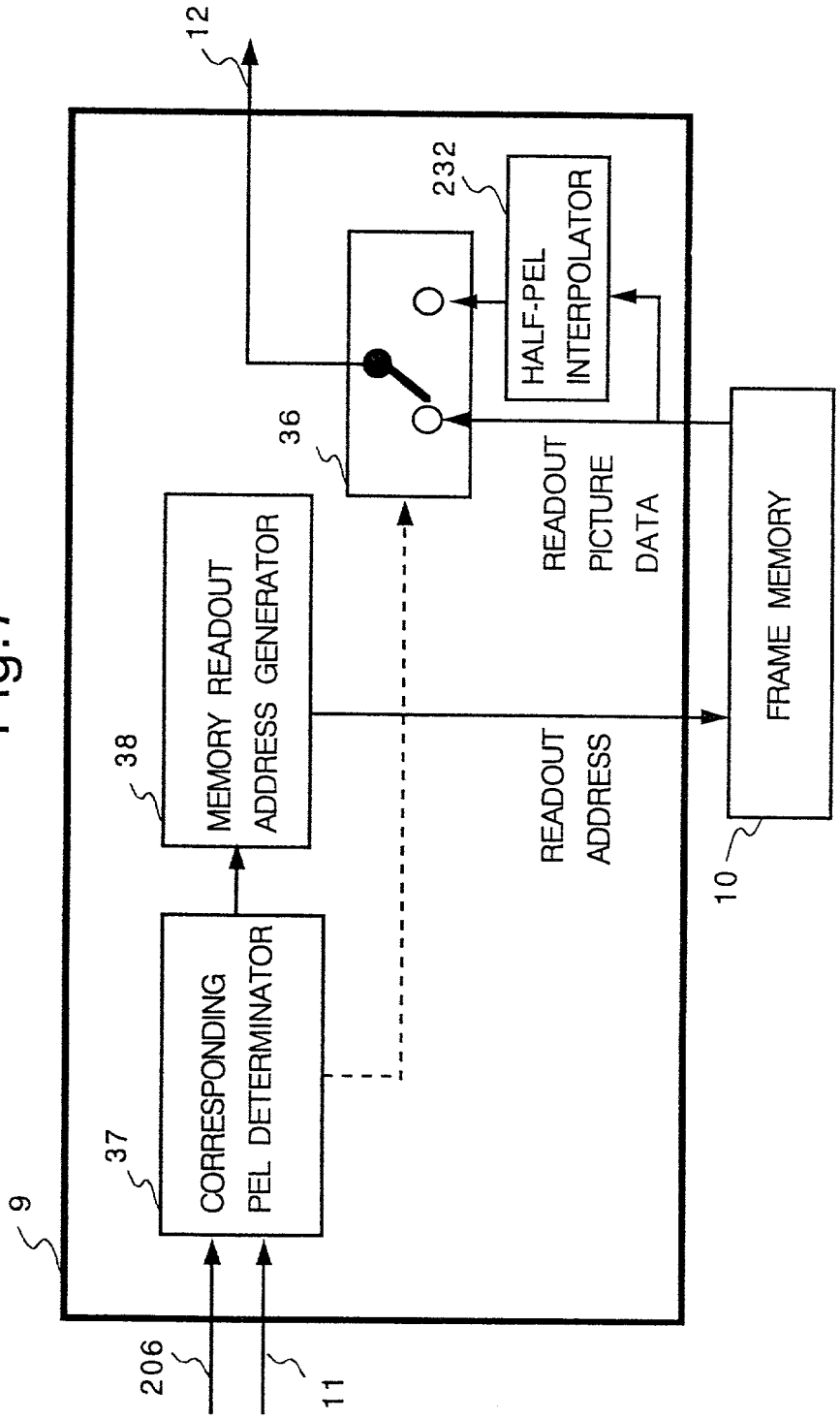

9
206
11
37
CORRESPONDING PEL DETERMINATOR
38
MEMORY READOUT ADDRESS GENERATOR
36
232
HALF-PEL INTERPOLATOR
12
READOUT ADDRESS
READOUT PICTURE DATA
10
FRAME MEMORY

Fig.8

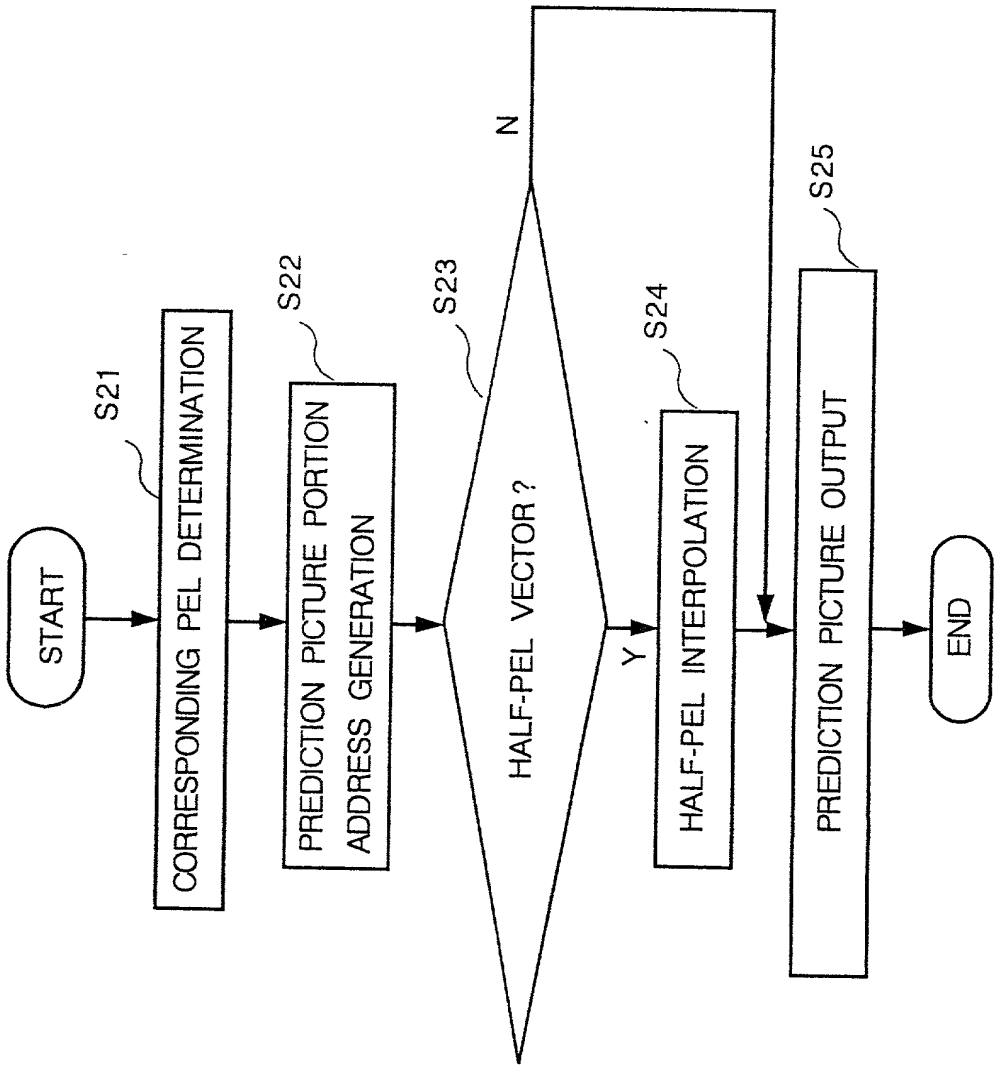

START
S21
CORRESPONDING PEL DETERMINATION
S22
PREDICTION PICTURE PORTION ADDRESS GENERATION
S23
HALF-PEL VECTOR ?
N
Y
S24
HALF-PEL INTERPOLATION
S25
PREDICTION PICTURE OUTPUT
END

Fig.9

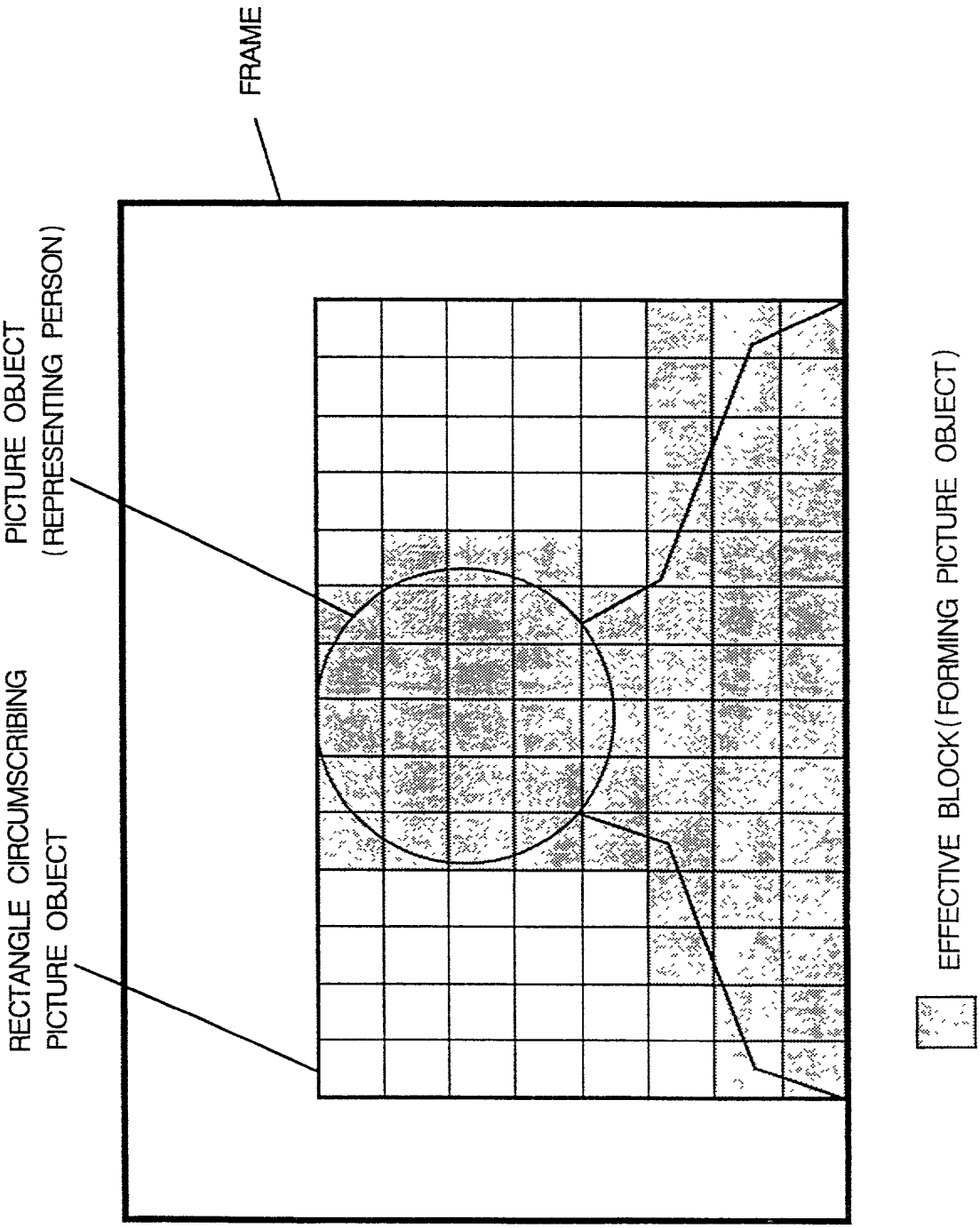

FRAME
PICTURE OBJECT
(REPRESENTING PERSON)
RECTANGLE CIRCUMSCRIBING
PICTURE OBJECT
EFFECTIVE BLOCK(FORMING PICTURE OBJECT)

Fig.10

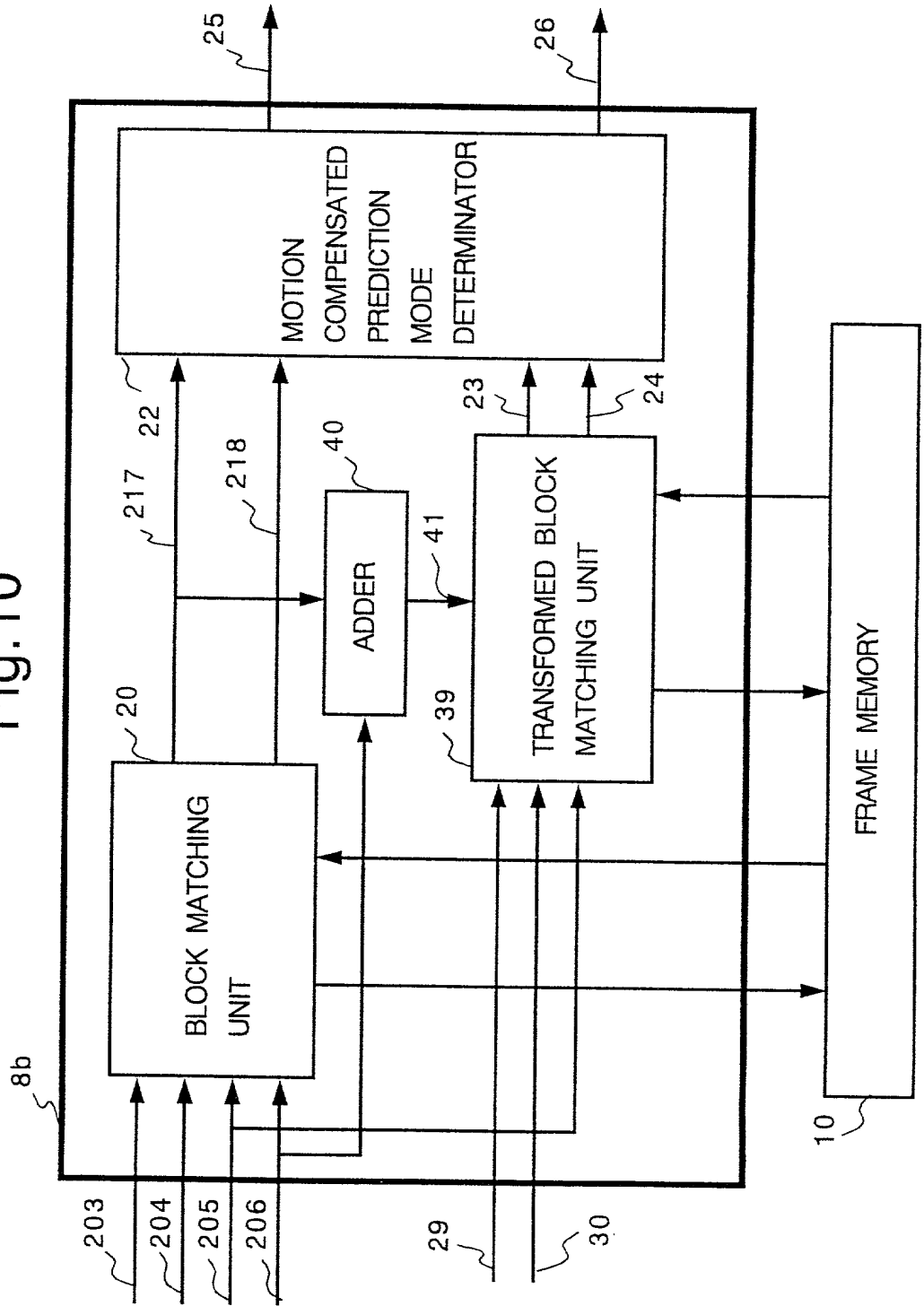

8b
BLOCK MATCHING UNIT
20
MOTION COMPENSATED PREDICTION MODE DETERMINATOR
22
ADDER
40
41
TRANSFORMED BLOCK MATCHING UNIT
23
24
39
FRAME MEMORY
10
25
26
217
218
203
204
205
206
29
30

Fig.11

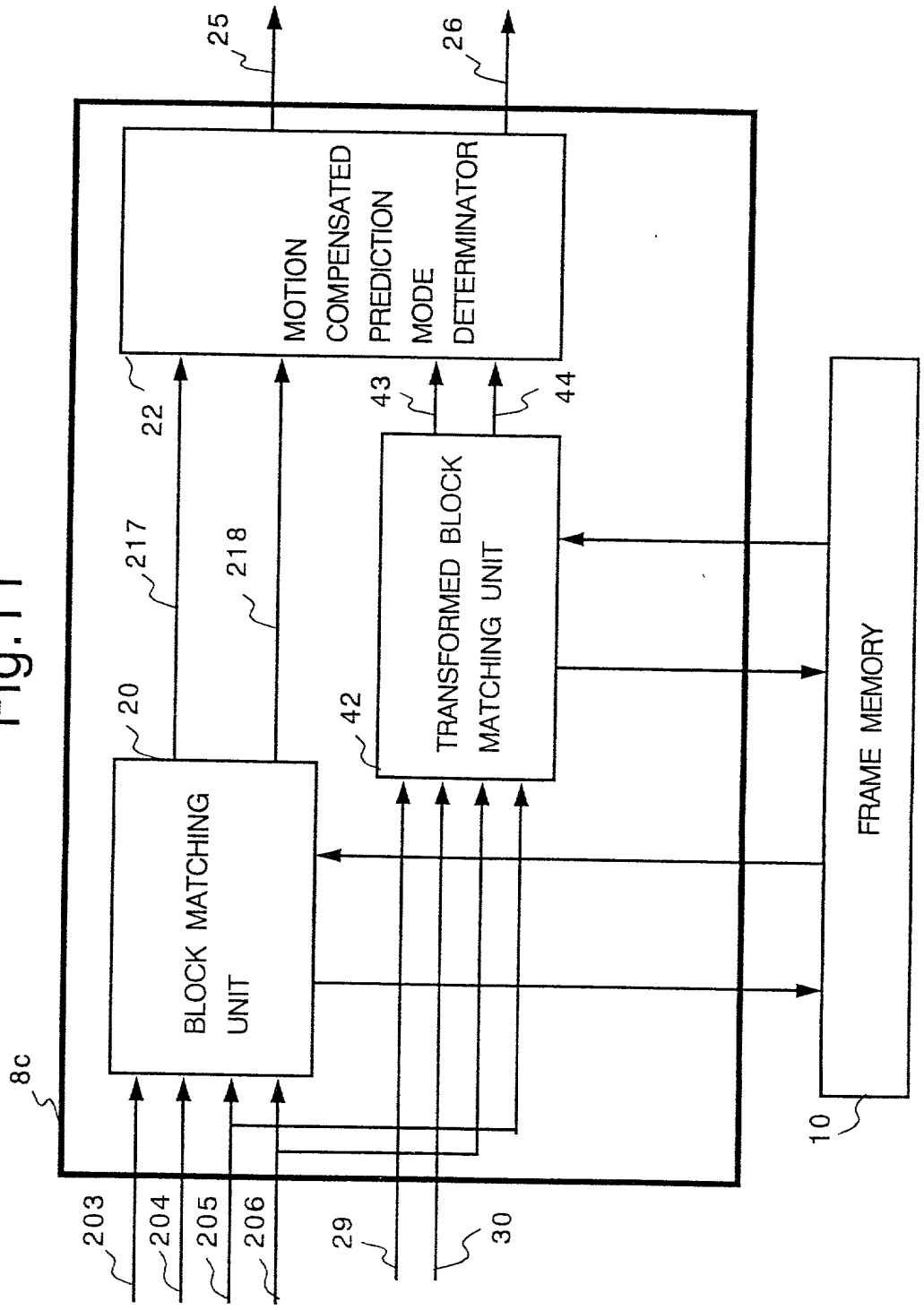

8c
BLOCK MATCHING UNIT
20
TRANSFORMED BLOCK MATCHING UNIT
42
MOTION COMPENSATED PREDICTION MODE DETERMINATOR
22
FRAME MEMORY
10
203
204
205
206
29
30
217
218
43
44
25
26

Fig.12

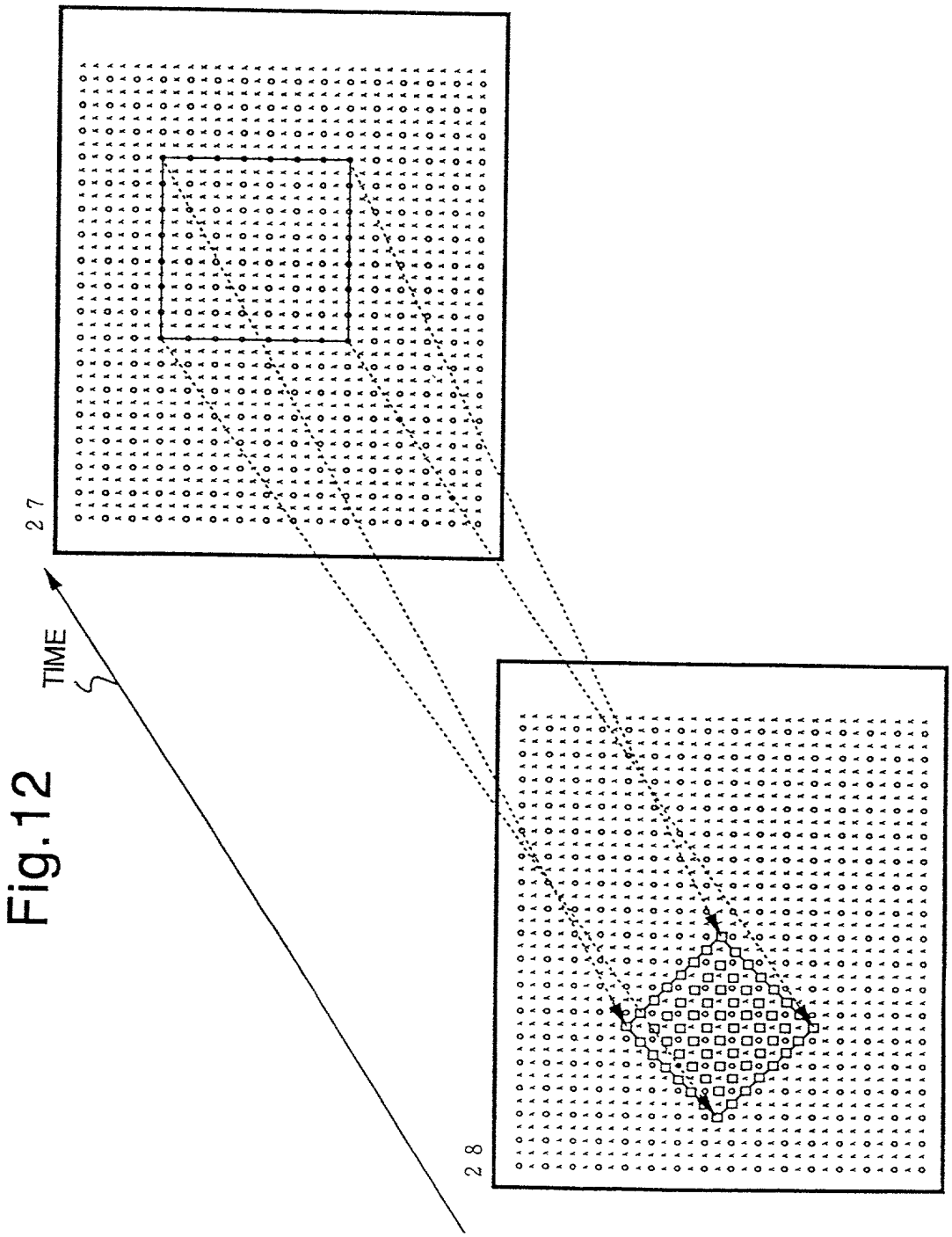


Fig.13

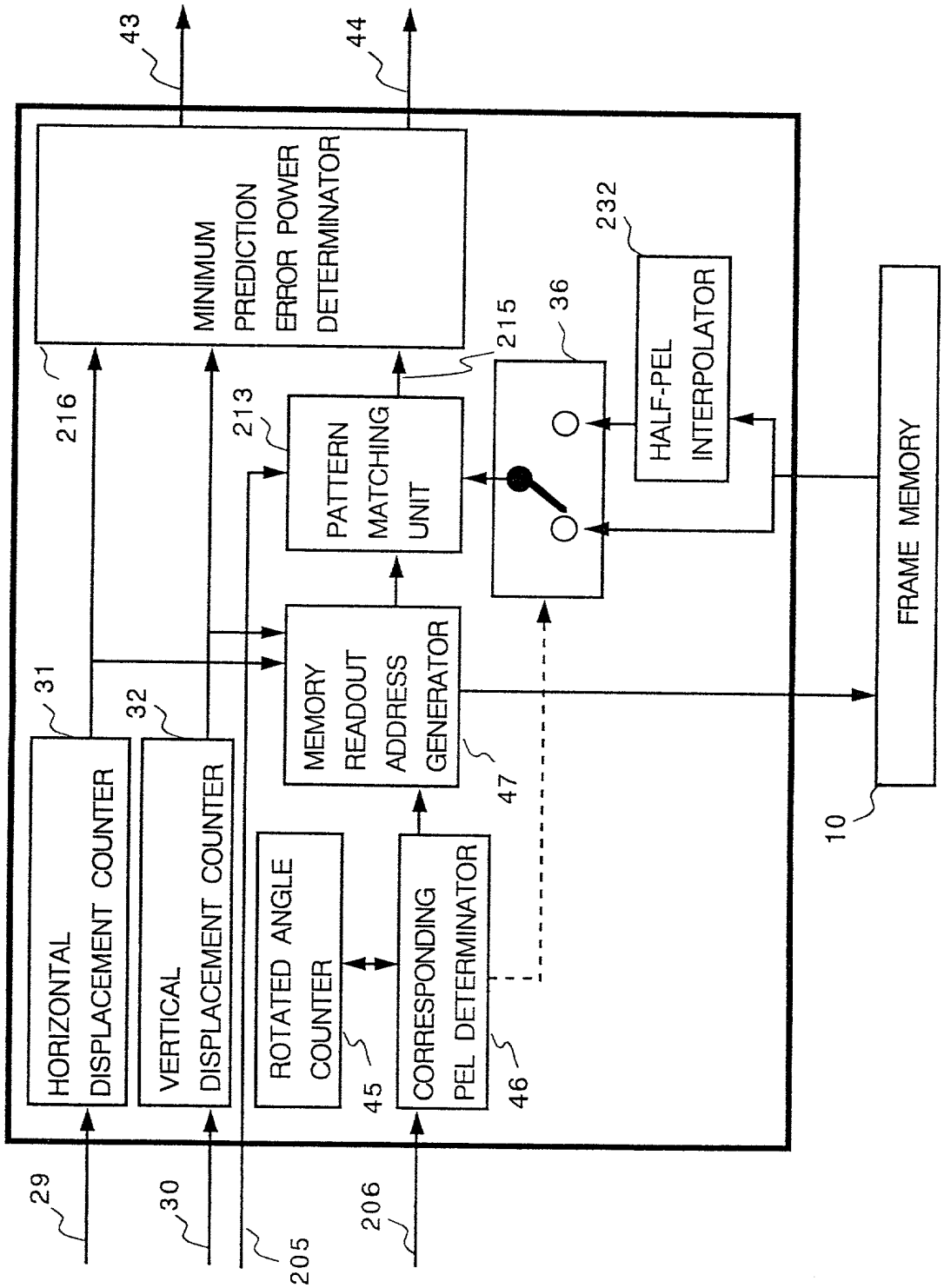

HORIZONTAL DISPLACEMENT COUNTER
VERTICAL DISPLACEMENT COUNTER
ROTATED ANGLE COUNTER
CORRESPONDING PEL DETERMINATOR
MEMORY READOUT ADDRESS GENERATOR
PATTERN MATCHING UNIT
MINIMUM PREDICTION ERROR POWER DETERMINATOR
HALF-PEL INTERPOLATOR
FRAME MEMORY
29
30
205
206
31
32
45
46
47
213
216
215
36
232
10
43
44

Fig.14

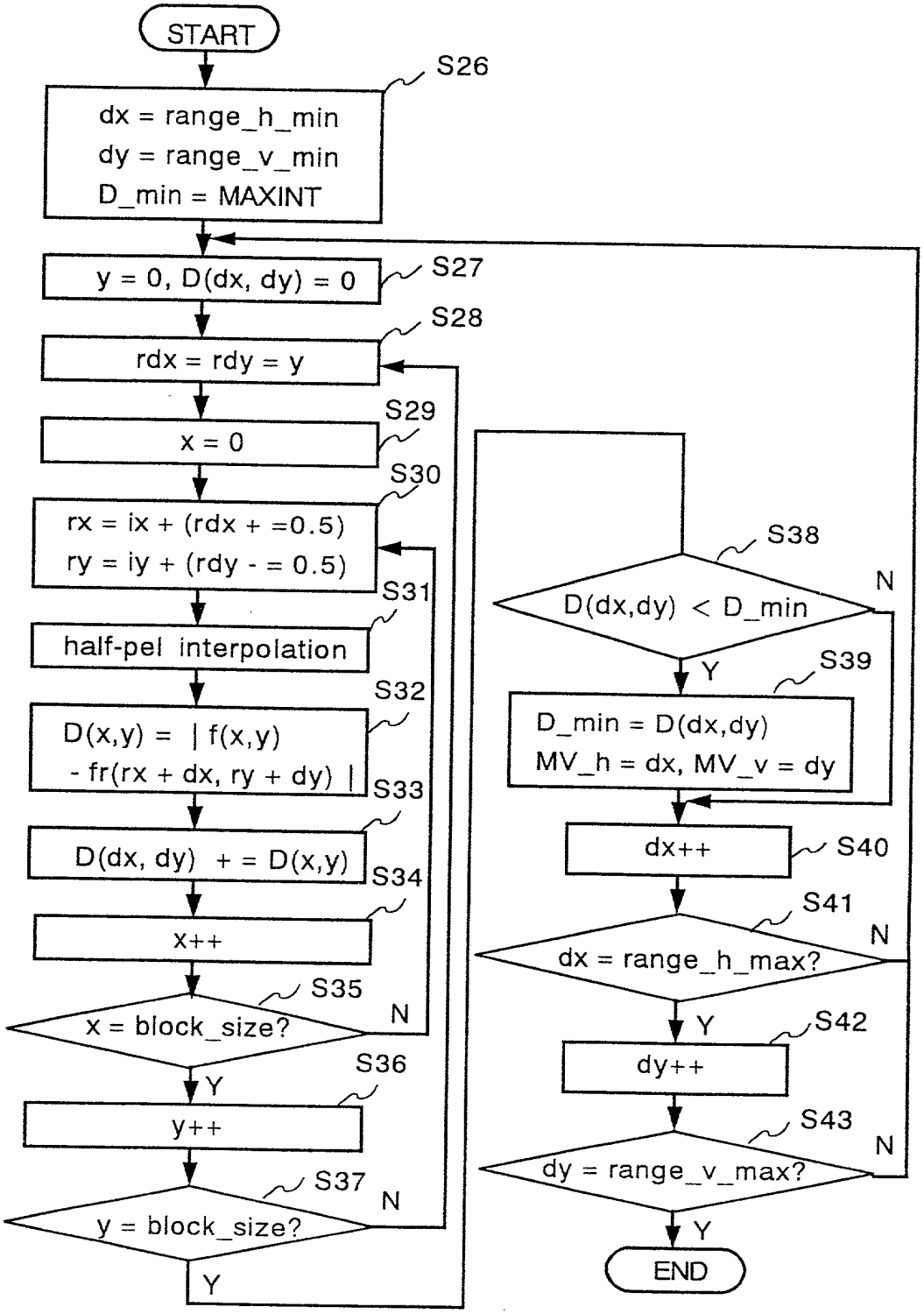

START
S26
dx = range_h_min
dy = range_v_min
D_min = MAXINT
y = 0, D(dx, dy) = 0
S27
S28
rdx = rdy = y
S29
x = 0
S30
rx = ix + (rdx + =0.5)
ry = iy + (rdy - = 0.5)
S31
half-pel interpolation
S32
D(x,y) = | f(x,y)
- fr(rx + dx, ry + dy) |
S33
D(dx, dy) + = D(x,y)
S34
x++
S35
x = block_size?
N
Y
S36
y++
S37
y = block_size?
N
Y
S38
D(dx,dy) < D_min
N
Y
S39
D_min = D(dx,dy)
MV_h = dx, MV_v = dy
dx++
S40
S41
dx = range_h_max?
N
Y
S42
dy++
S43
dy = range_v_max?
N
Y
END

Fig.15

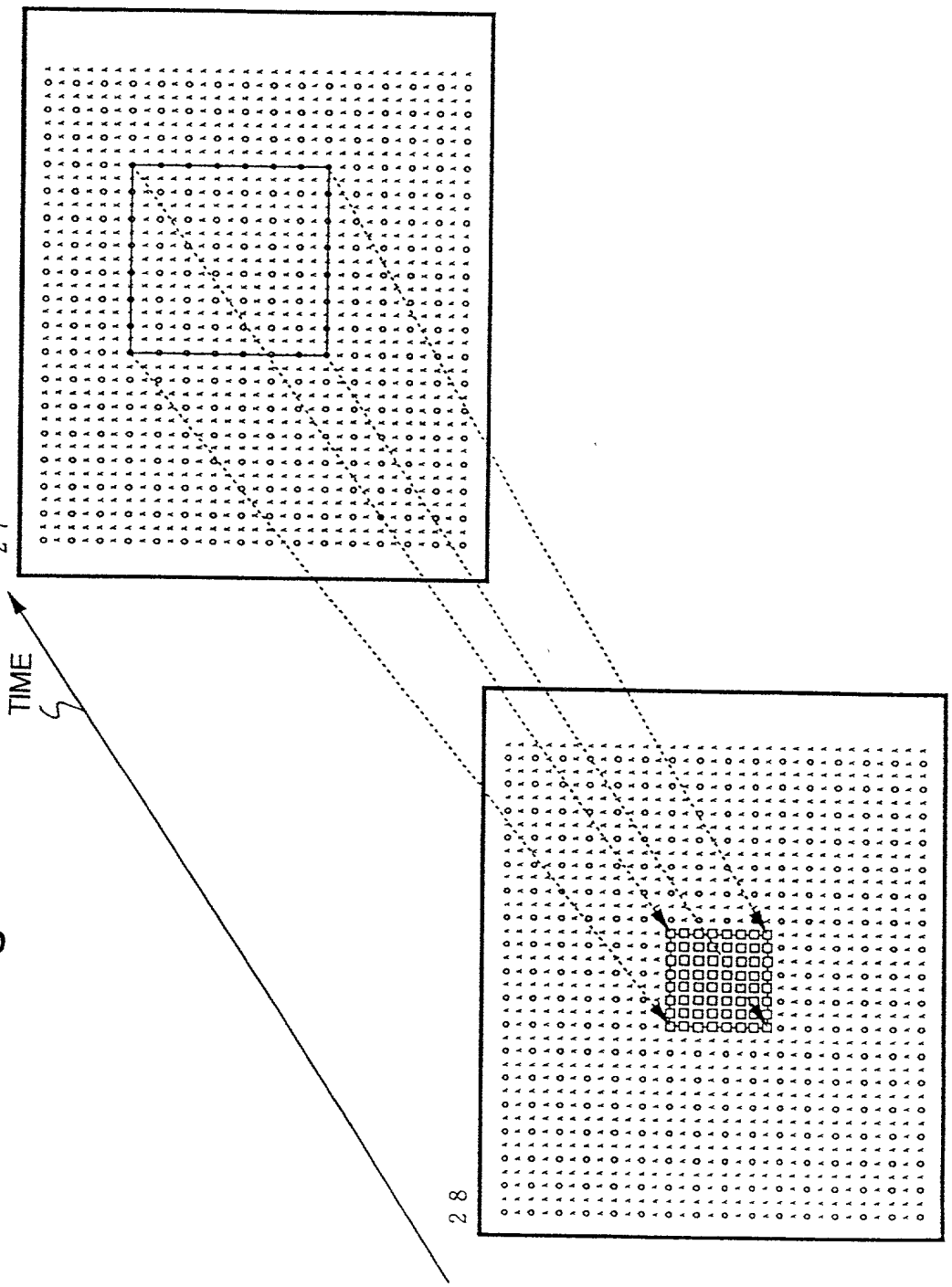


Fig.16

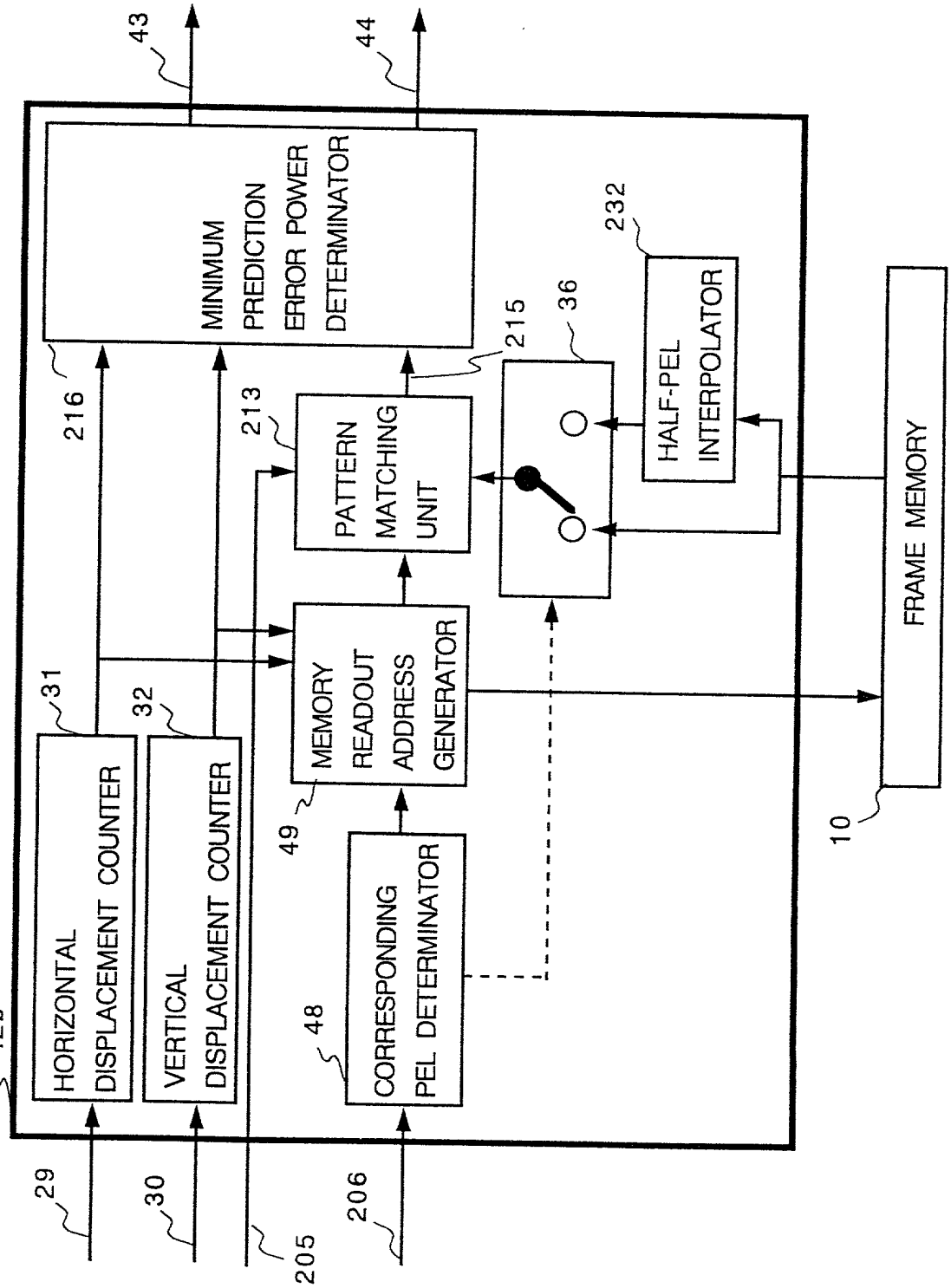

42b
HORIZONTAL DISPLACEMENT COUNTER
VERTICAL DISPLACEMENT COUNTER
MEMORY READOUT ADDRESS GENERATOR
CORRESPONDING PEL DETERMINATOR
PATTERN MATCHING UNIT
MINIMUM PREDICTION ERROR POWER DETERMINATOR
HALF-PEL INTERPOLATOR
FRAME MEMORY
29
30
205
206
31
32
49
48
213
216
215
36
232
43
44
10

Fig.17

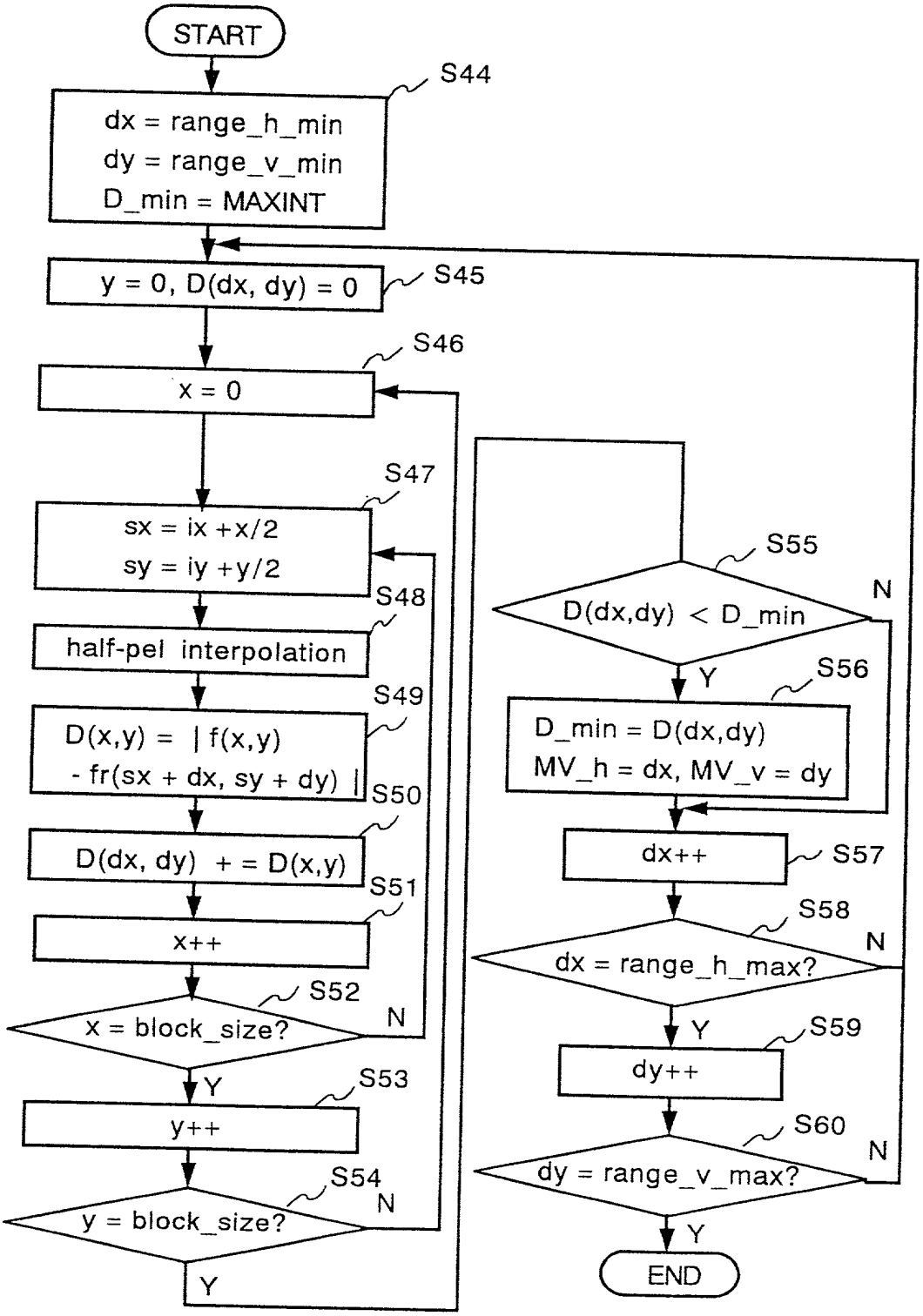

START
S44
dx = range_h_min
dy = range_v_min
D_min = MAXINT
y = 0, D(dx, dy) = 0
S45
S46
x = 0
S47
sx = ix +x/2
sy = iy +y/2
S48
half-pel interpolation
S49
D(x,y) = | f(x,y)
- fr(sx + dx, sy + dy) |
S50
D(dx, dy) + = D(x,y)
S51
x++
S52
x = block_size?
N
Y
S53
y++
S54
y = block_size?
N
Y
S55
D(dx,dy) < D_min
N
Y
S56
D_min = D(dx,dy)
MV_h = dx, MV_v = dy
dx++
S57
S58
dx = range_h_max?
N
Y
S59
dy++
S60
dy = range_v_max?
N
Y
END

Fig.18

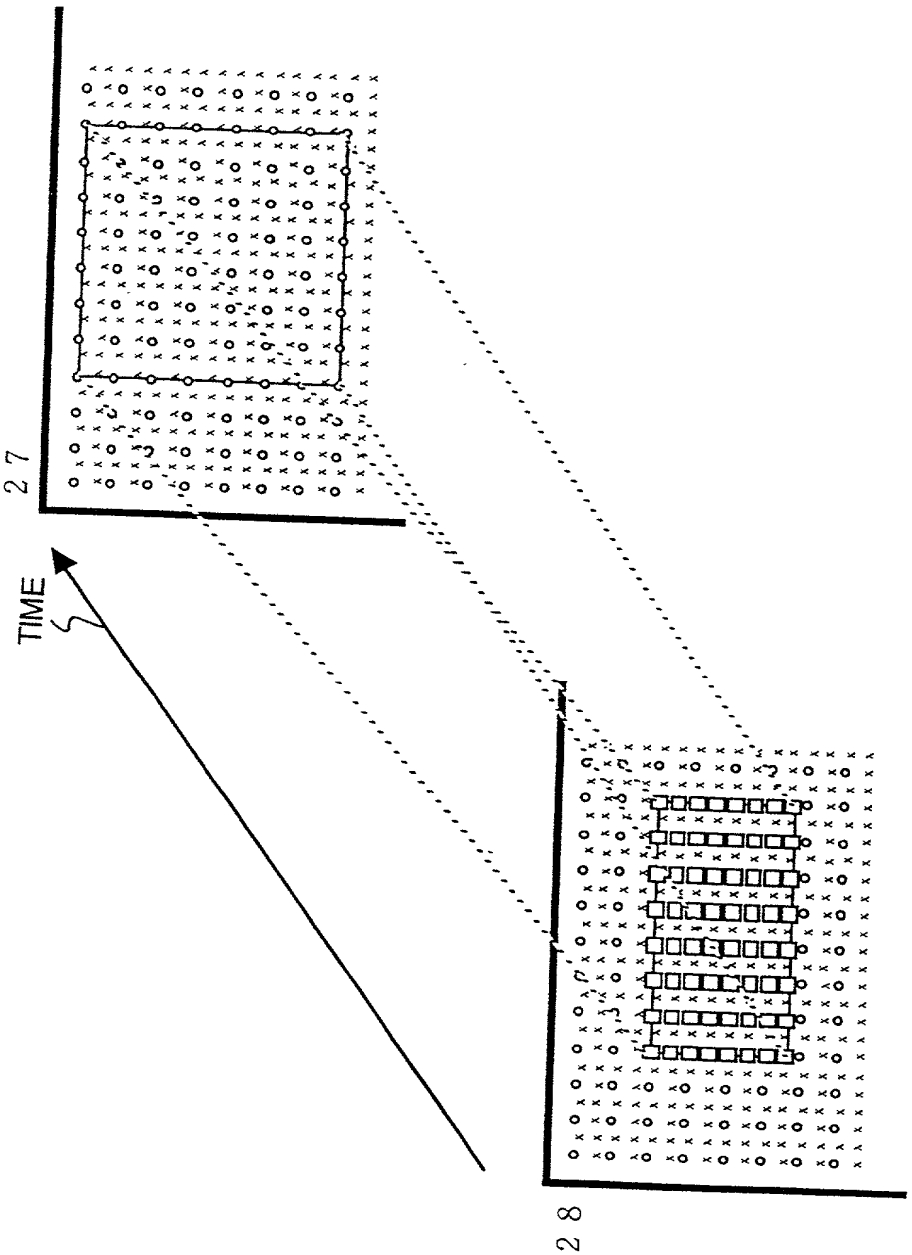


Fig.19

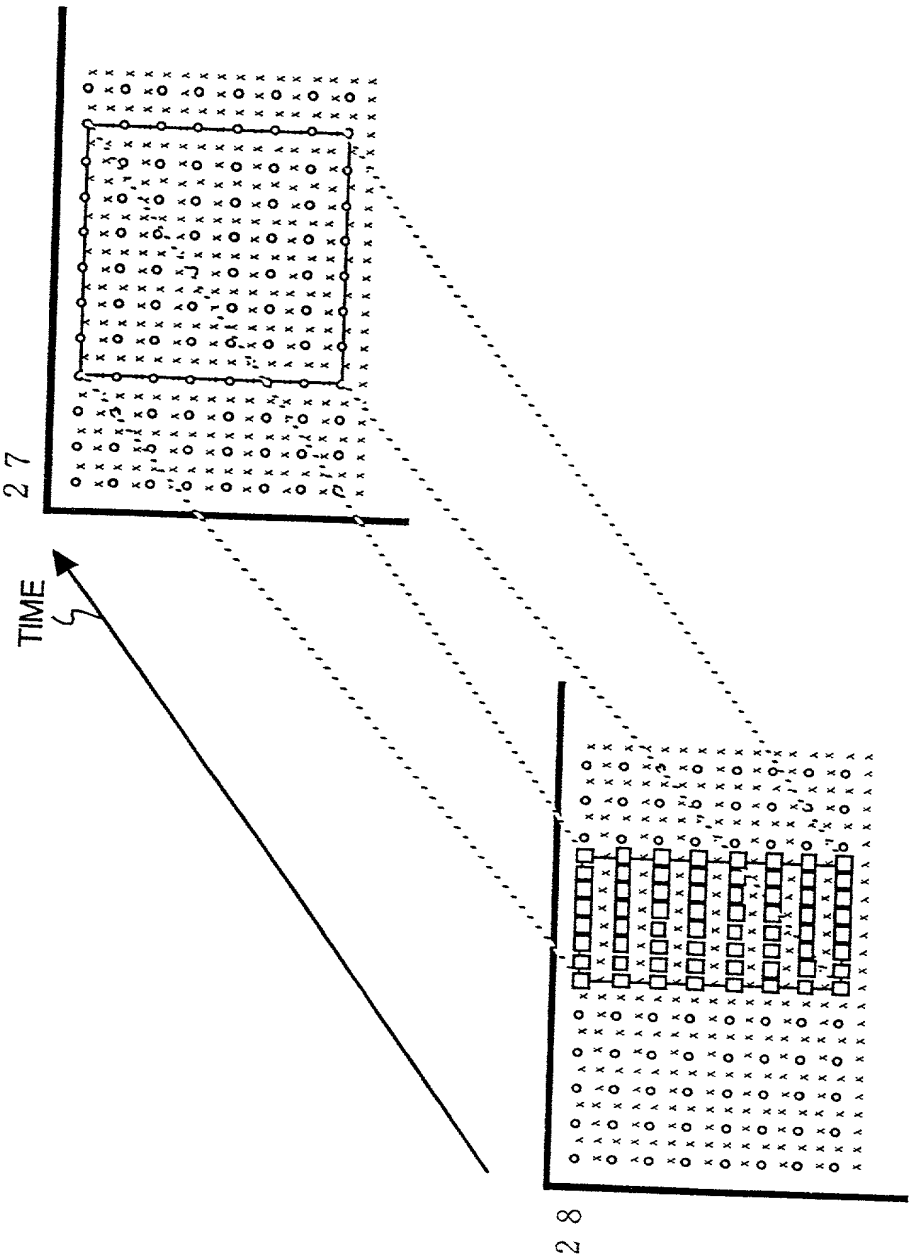


Fig.20

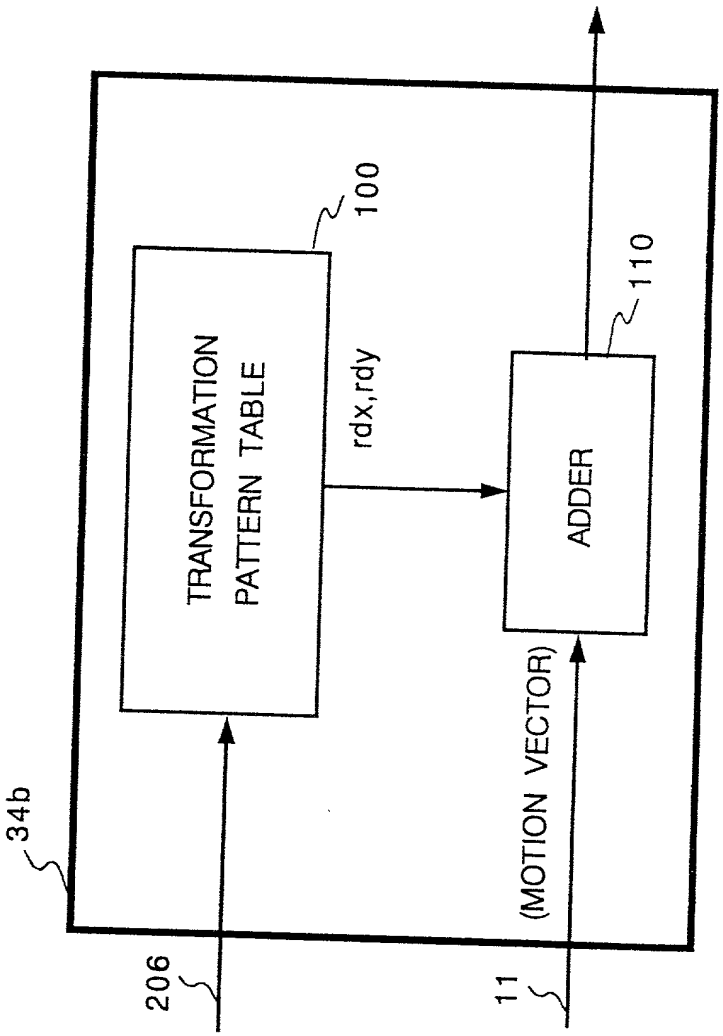

34b
206
11
TRANSFORMATION PATTERN TABLE
100
rdx,rdy
ADDER
110
(MOTION VECTOR)

Fig.21

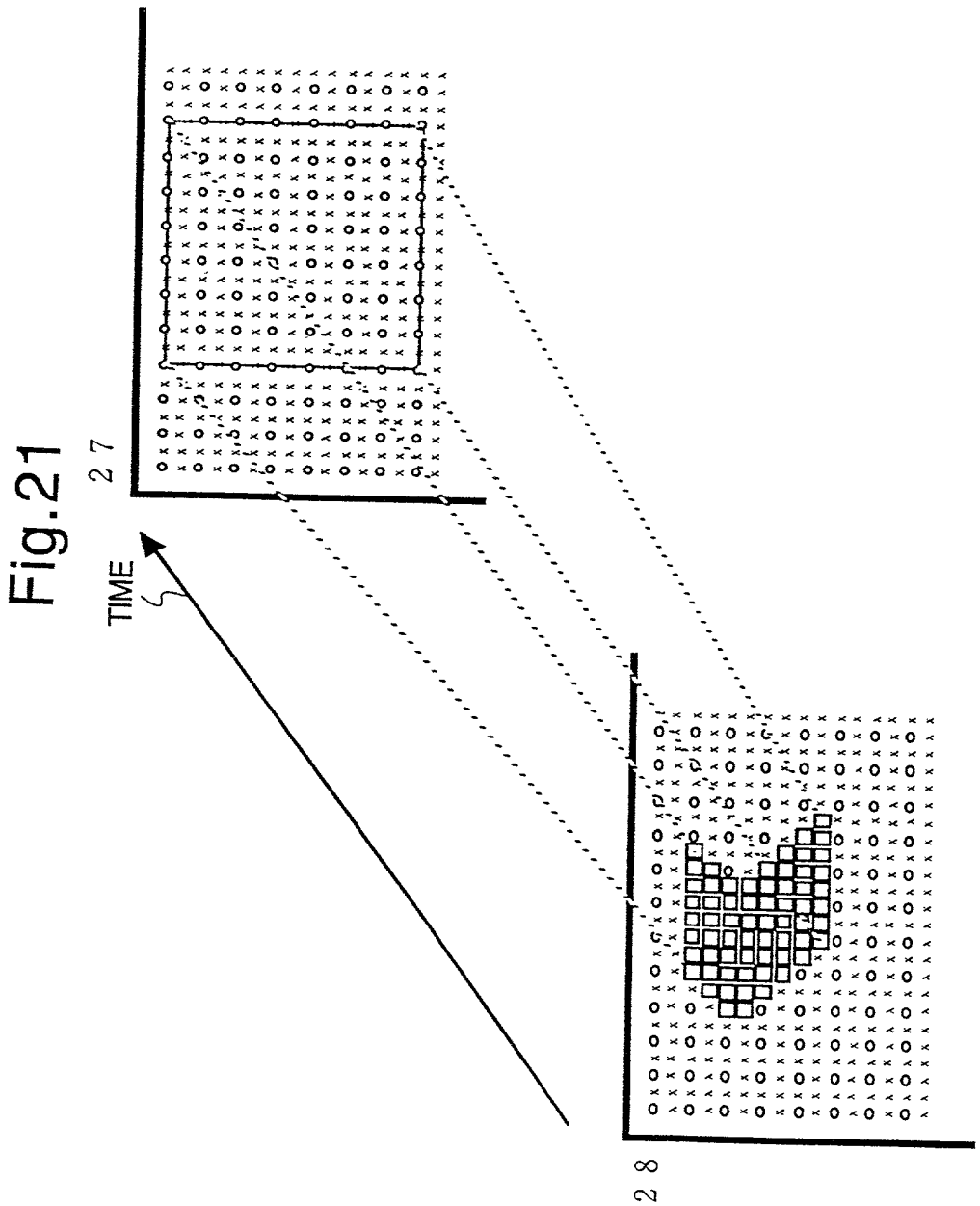


Fig.22

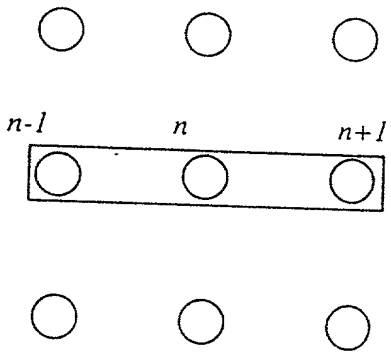
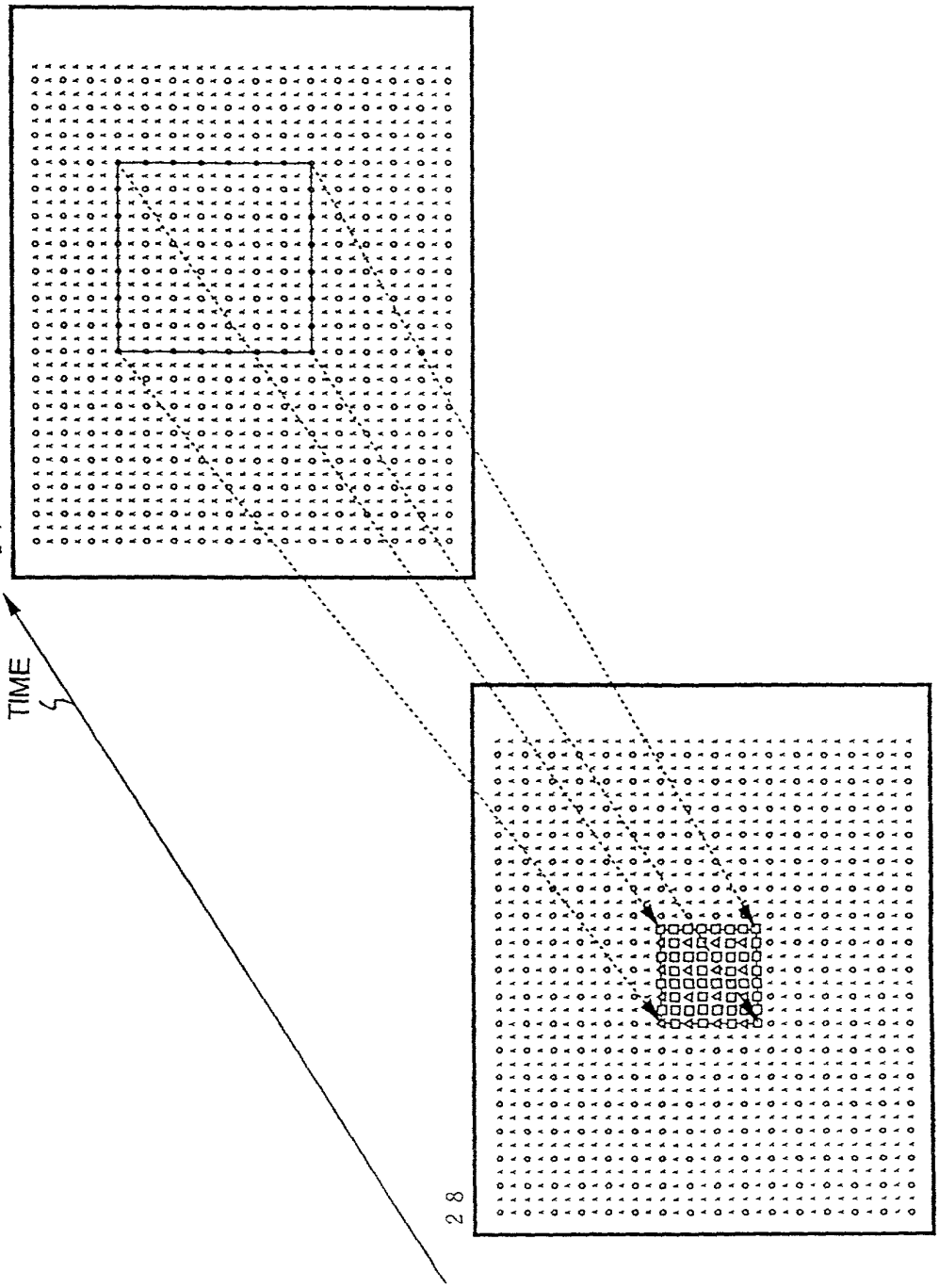


Fig.23



20180207 Shift 330 01

Fig.24

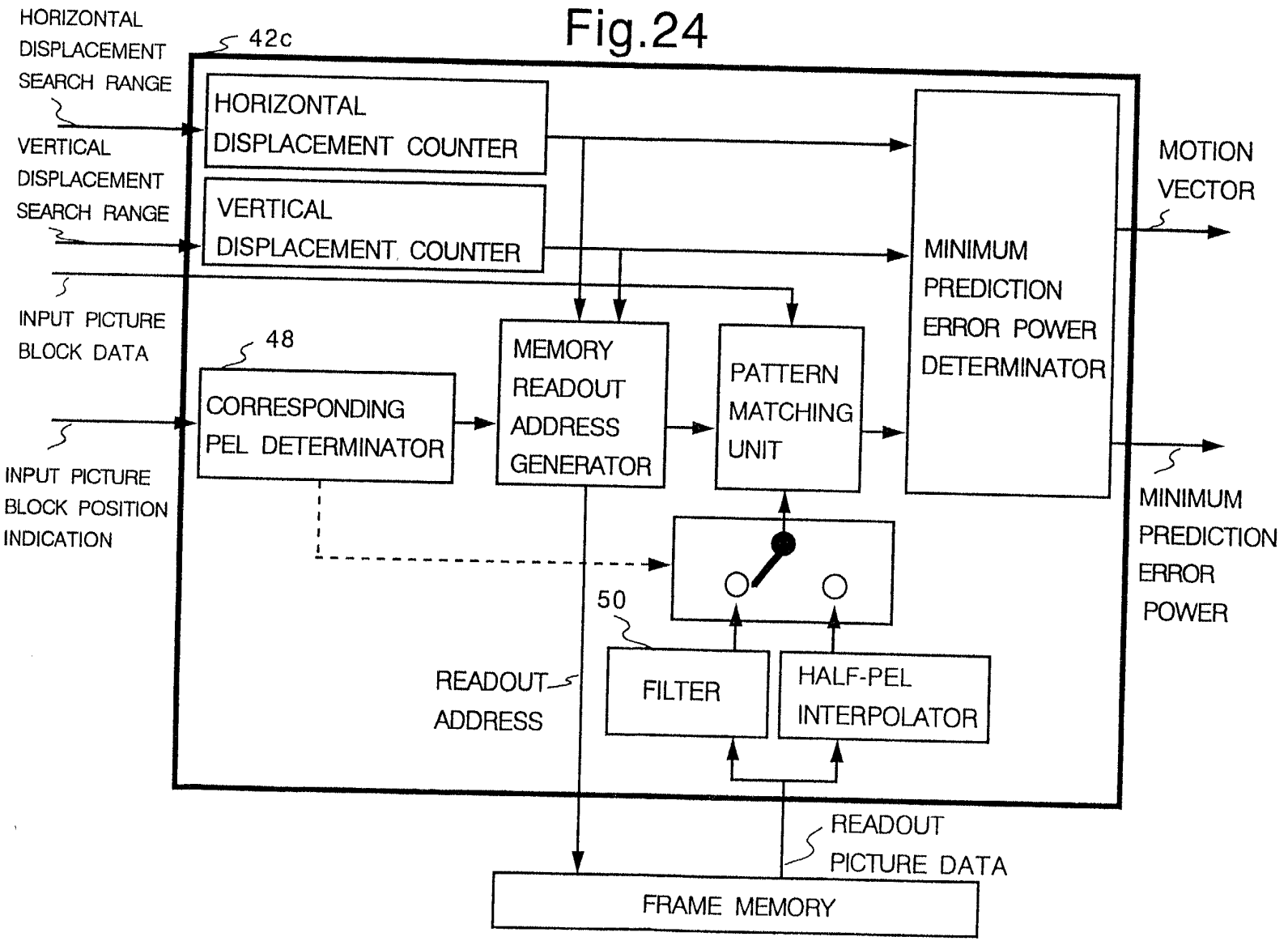


Fig.25

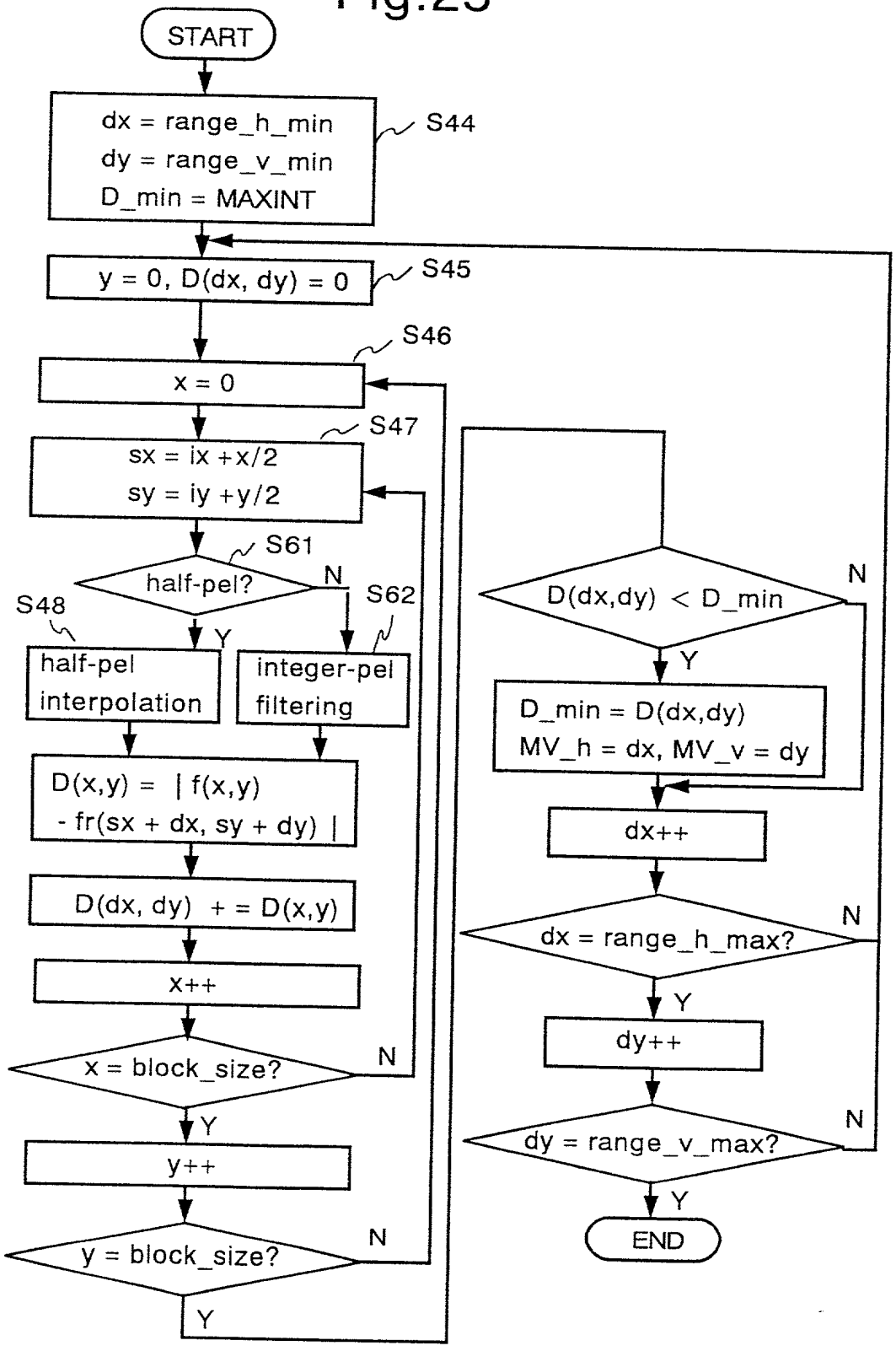

START
dx = range_h_min
dy = range_v_min
D_min = MAXINT
S44
y = 0, D(dx, dy) = 0
S45
S46
x = 0
S47
sx = ix +x/2
sy = iy +y/2
S61
half-pel?
N
Y
S48
S62
half-pel interpolation
integer-pel filtering
D(x,y) = | f(x,y) - fr(sx + dx, sy + dy) |
D(dx, dy) + = D(x,y)
x++
x = block_size?
N
Y
y++
y = block_size?
N
Y
D(dx,dy) < D_min
N
Y
D_min = D(dx,dy)
MV_h = dx, MV_v = dy
dx++
dx = range_h_max?
N
Y
dy++
dy = range_v_max?
N
Y
END

Fig.26

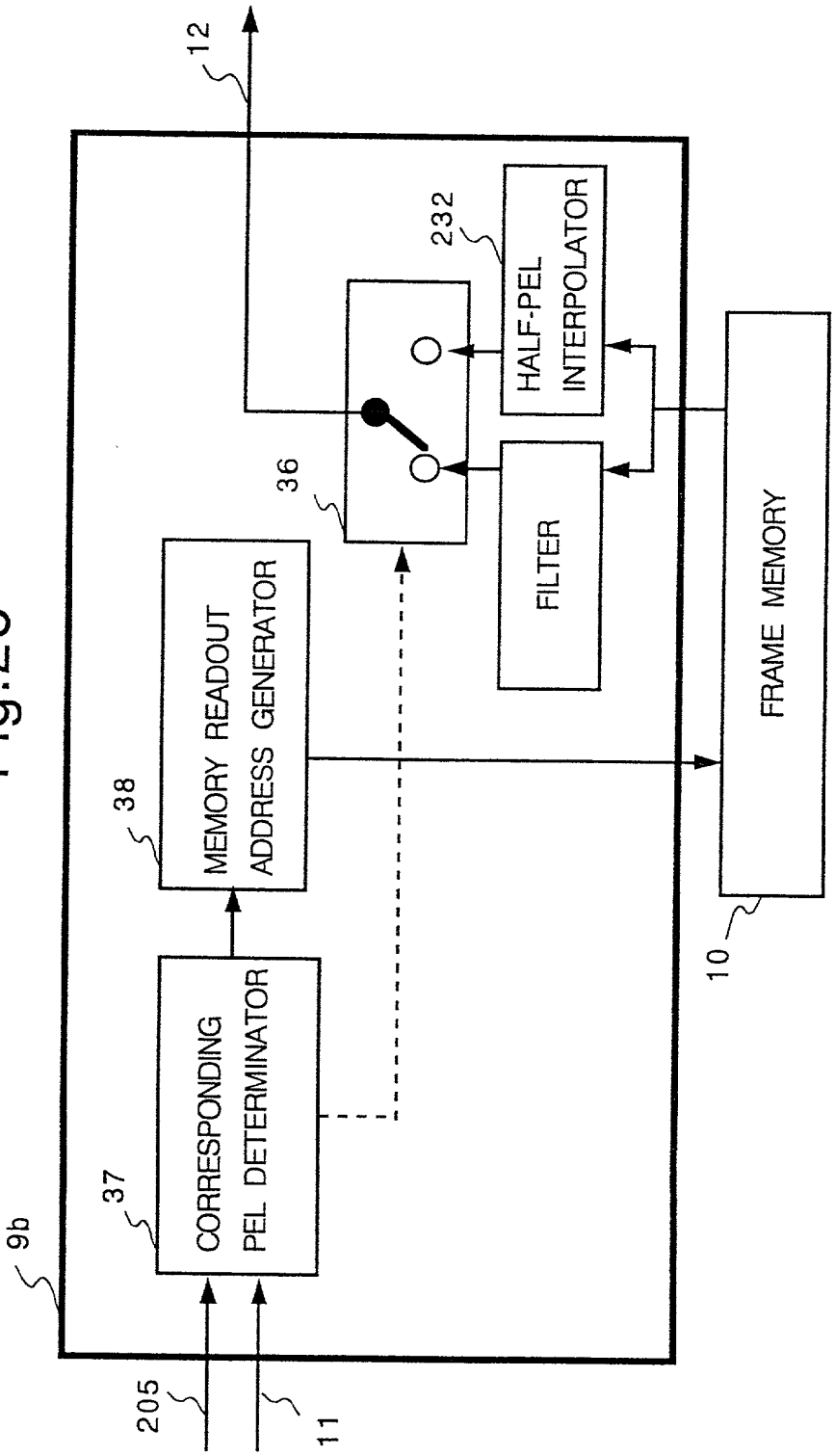

12
232
HALF-PEL INTERPOLATOR
36
FILTER
FRAME MEMORY
10
38
MEMORY READOUT ADDRESS GENERATOR
37
CORRESPONDING PEL DETERMINATOR
9b
205
11

Fig.27

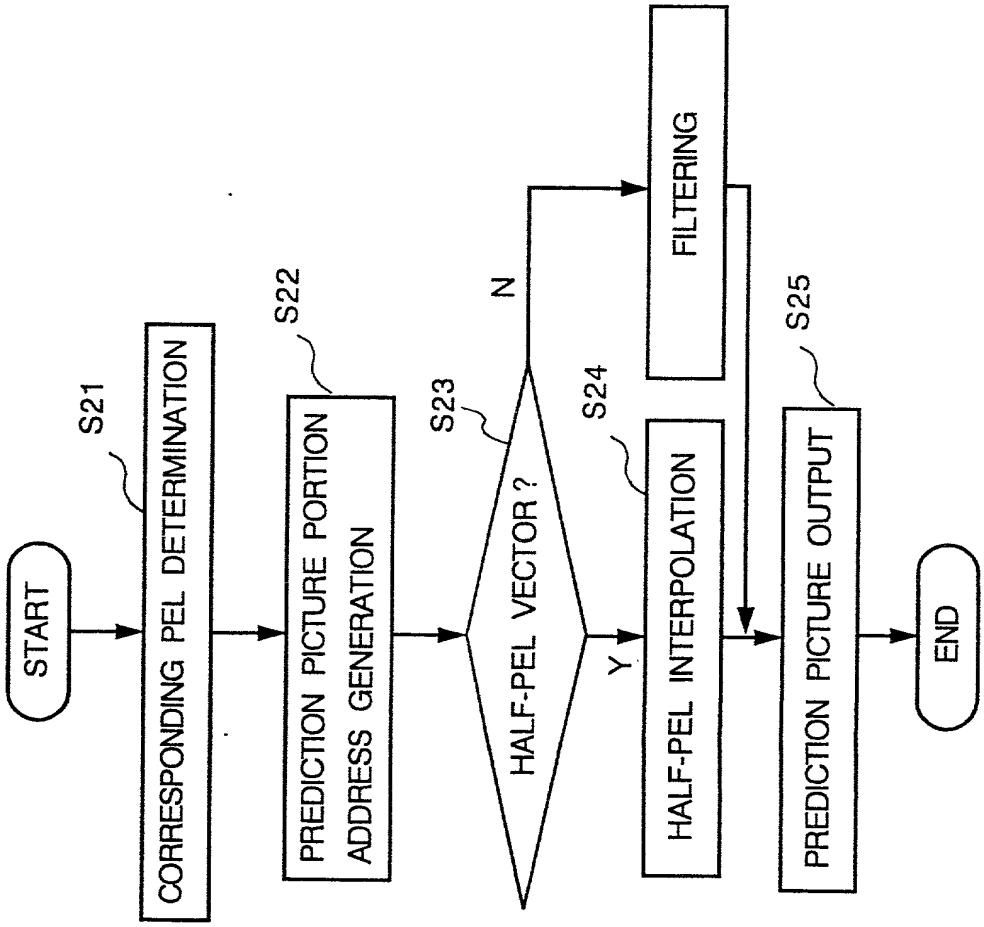

START
S21
CORRESPONDING PEL DETERMINATION
S22
PREDICTION PICTURE PORTION ADDRESS GENERATION
S23
HALF-PEL VECTOR ?
N
Y
S24
FILTERING
HALF-PEL INTERPOLATION
S25
PREDICTION PICTURE OUTPUT
END

Fig.28

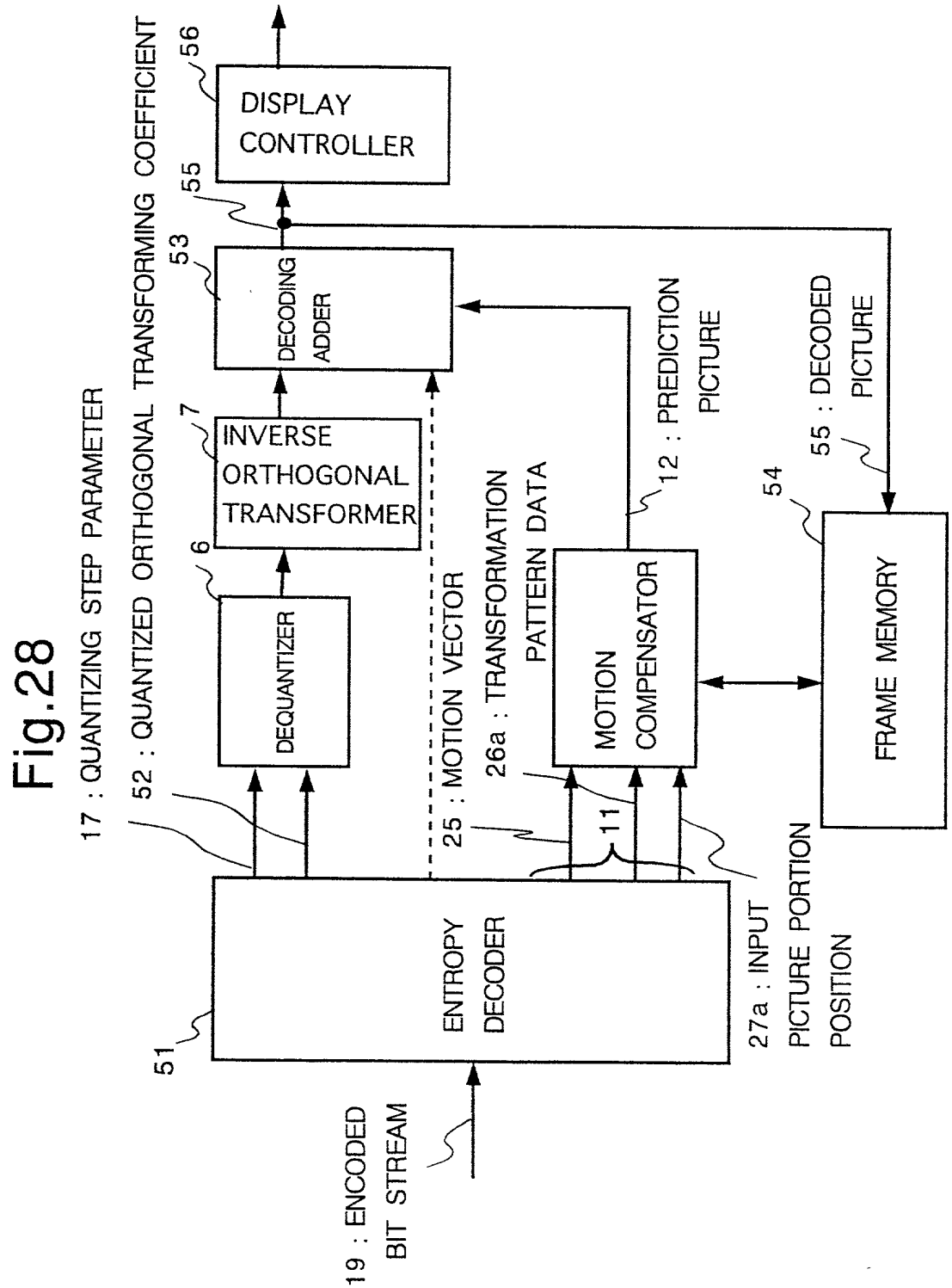

17 : QUANTIZING STEP PARAMETER
52 : QUANTIZED ORTHOGONAL TRANSFORMING COEFFICIENT
25 : MOTION VECTOR
26a : TRANSFORMATION PATTERN DATA
12 : PREDICTION PICTURE
55 : DECODED PICTURE
27a : INPUT PICTURE PORTION POSITION
19 : ENCODED BIT STREAM
51
ENTROPY DECODER
6
DEQUANTIZER
7
INVERSE ORTHOGONAL TRANSFORMER
53
DECODING ADDER
55
56
DISPLAY CONTROLLER
11
MOTION COMPENSATOR
54
FRAME MEMORY

Fig.29

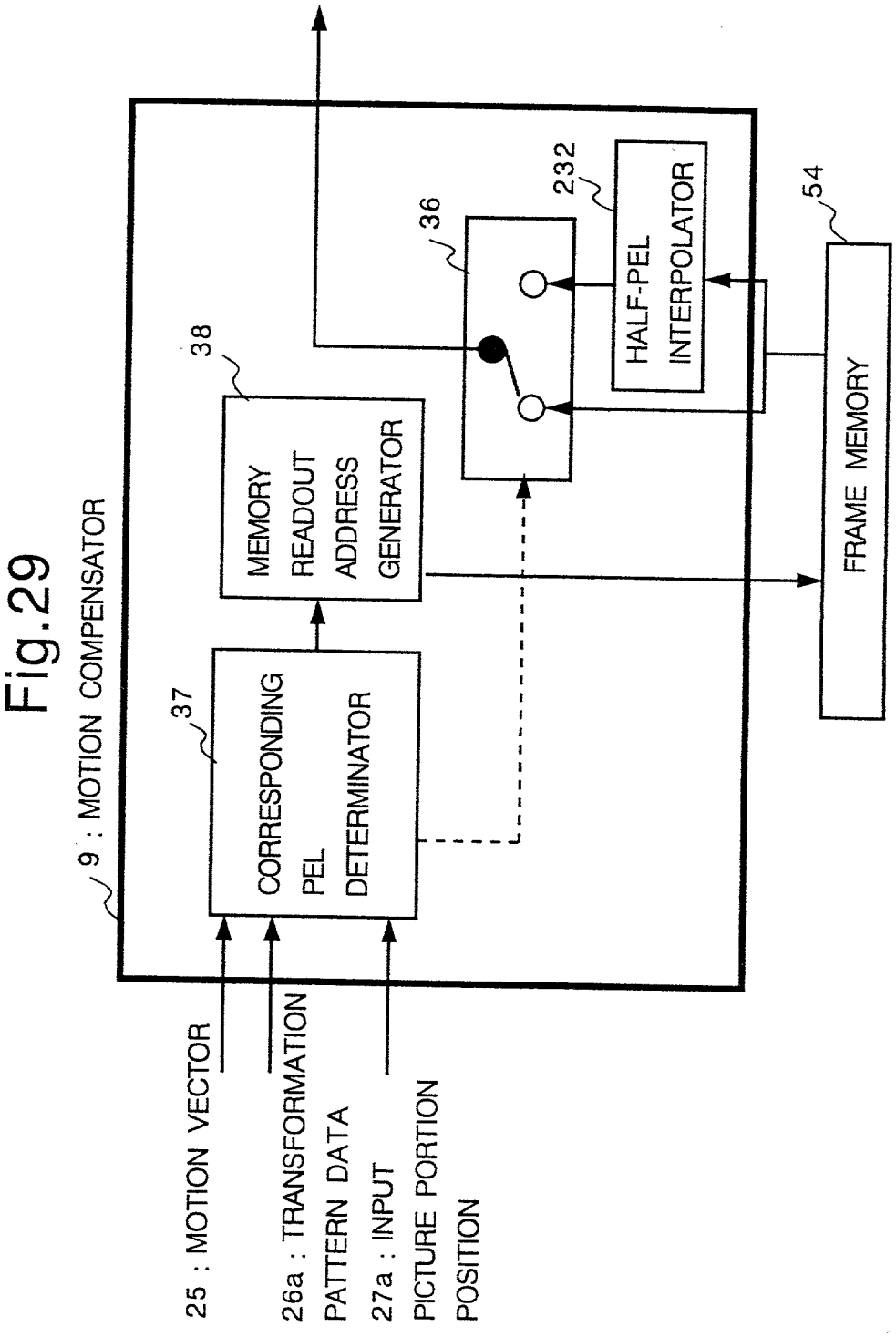

9 : MOTION COMPENSATOR
37
CORRESPONDING PEL DETERMINATOR
38
MEMORY READOUT ADDRESS GENERATOR
36
232
HALF-PEL INTERPOLATOR
54
FRAME MEMORY
25 : MOTION VECTOR
26a : TRANSFORMATION PATTERN DATA
27a : INPUT PICTURE PORTION POSITION

Fig.30

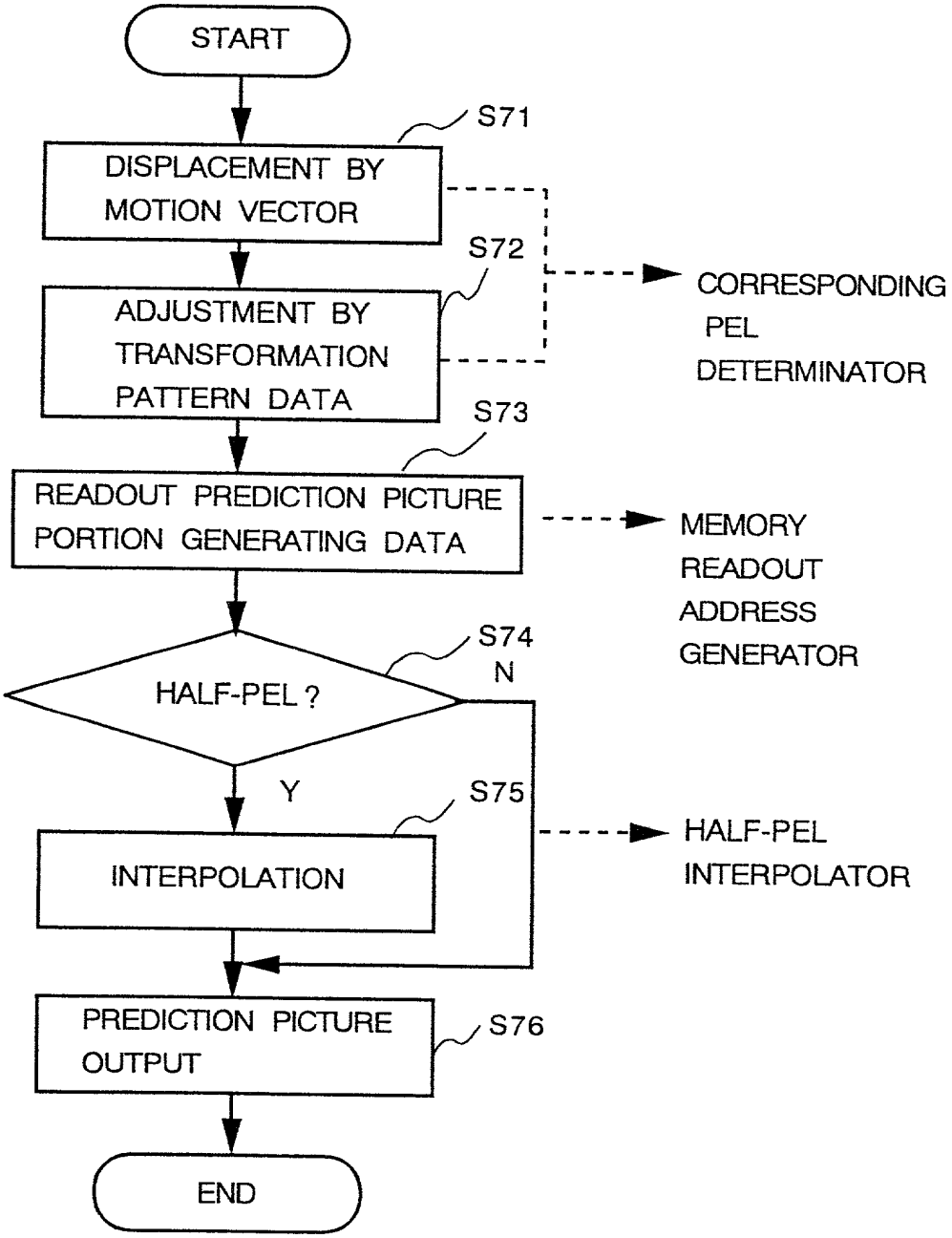

START
S71
DISPLACEMENT BY MOTION VECTOR
S72
ADJUSTMENT BY TRANSFORMATION PATTERN DATA
CORRESPONDING PEL DETERMINATOR
S73
READOUT PREDICTION PICTURE PORTION GENERATING DATA
MEMORY READOUT ADDRESS GENERATOR
S74
HALF-PEL ?
N
Y
S75
INTERPOLATION
HALF-PEL INTERPOLATOR
PREDICTION PICTURE OUTPUT
S76
END

Fig.31

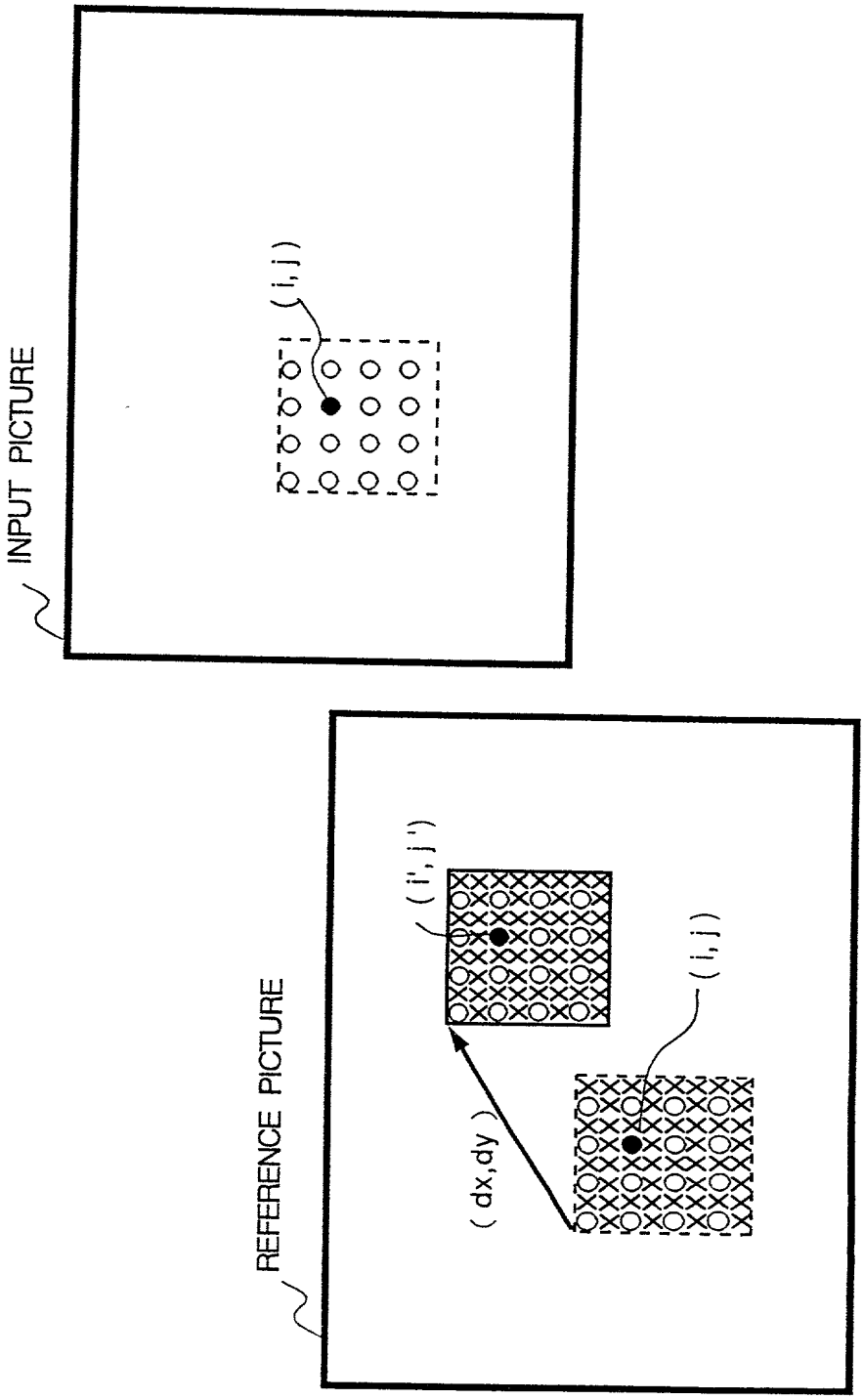


Fig.32

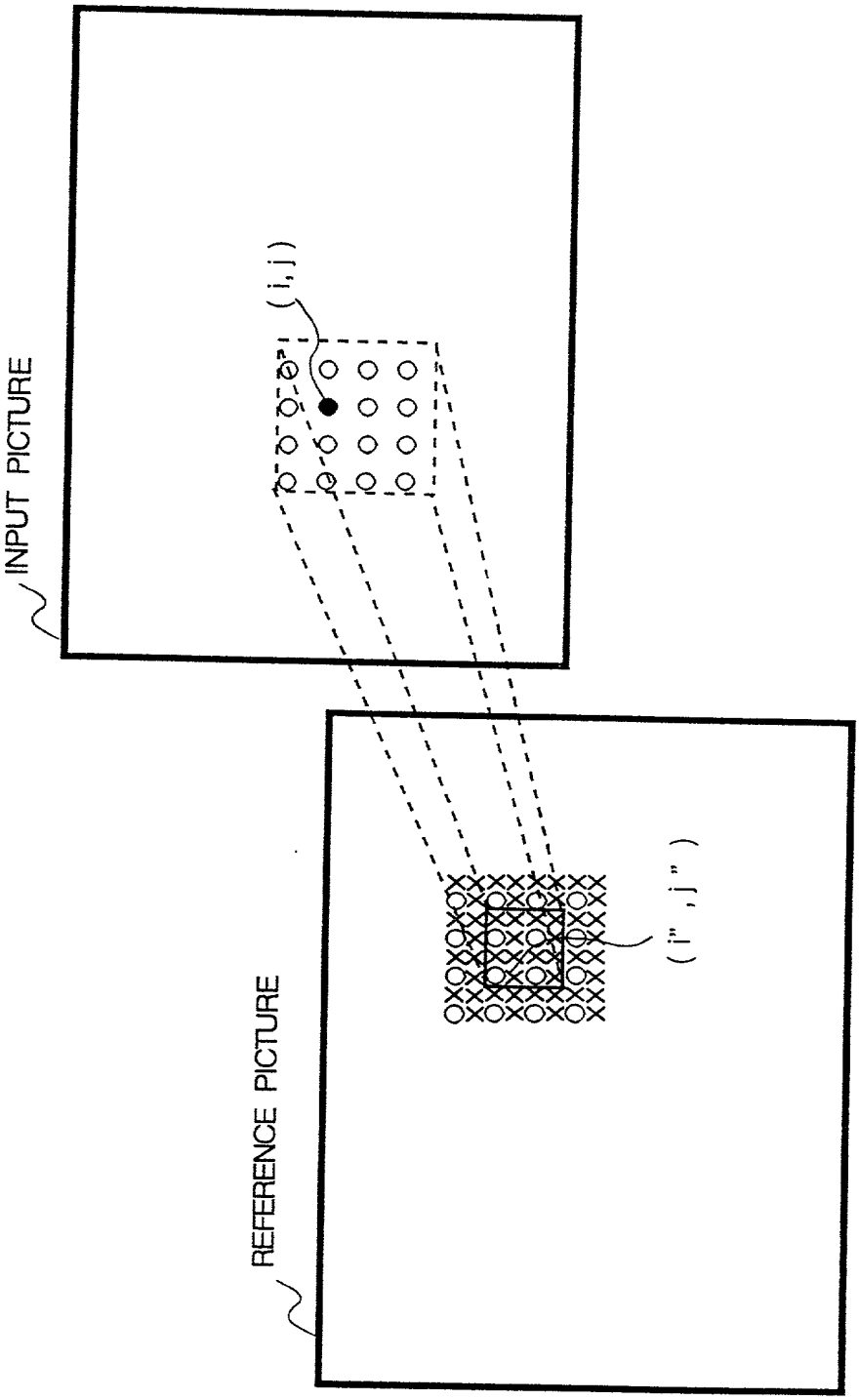
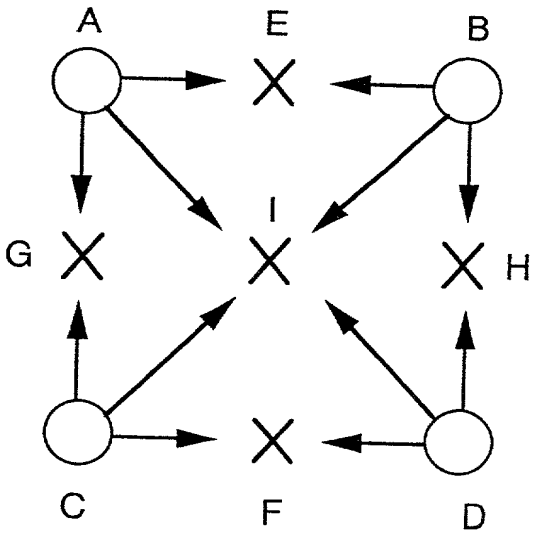


Fig.33



A,B,C,D : INTEGER PEL

E=(A+B) / 2
F=(C+D) / 2
G=(A+C) / 2
H=(B+D) / 2
EI=(A+B+C+D) / 4

Fig.34

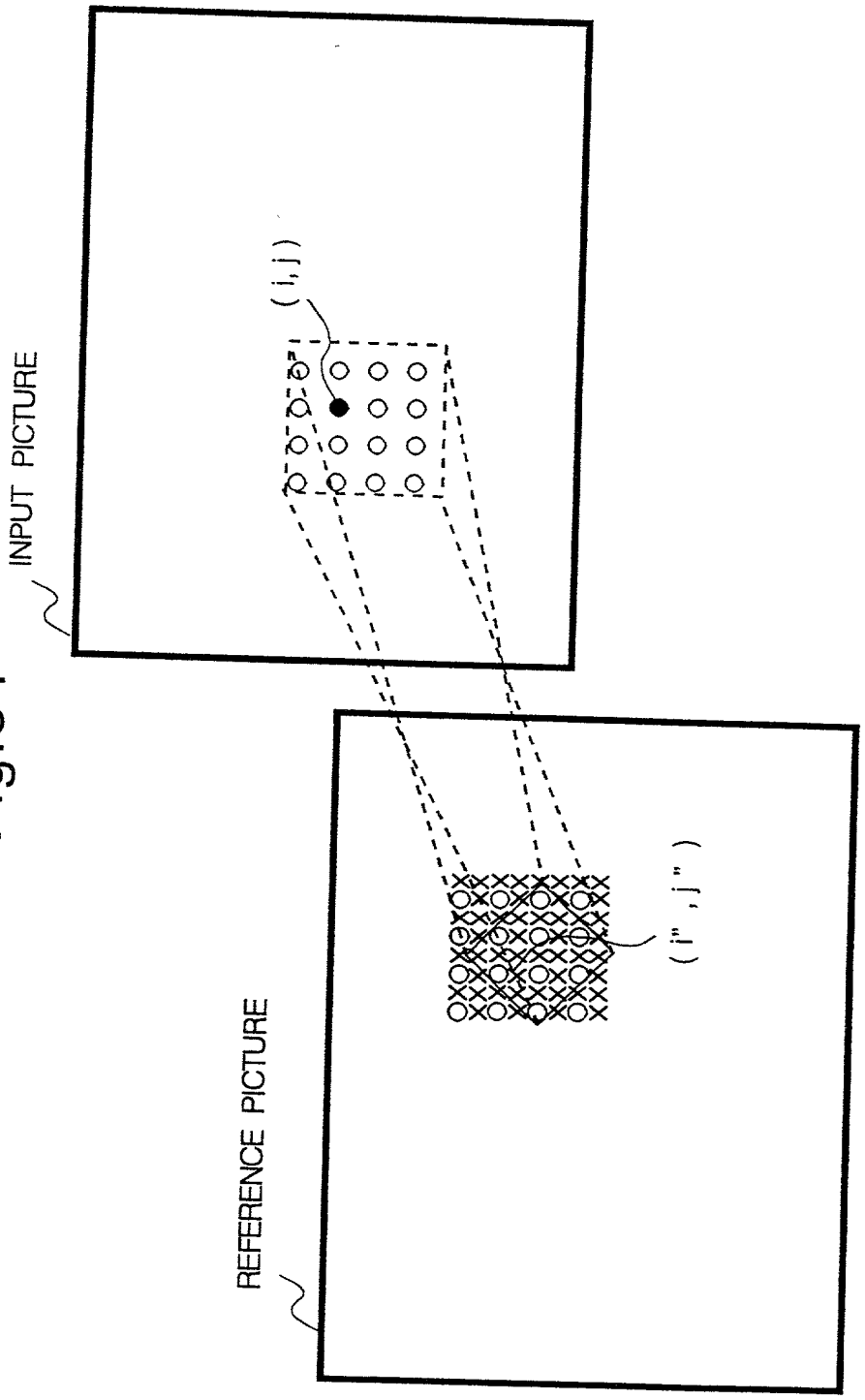


Fig.35

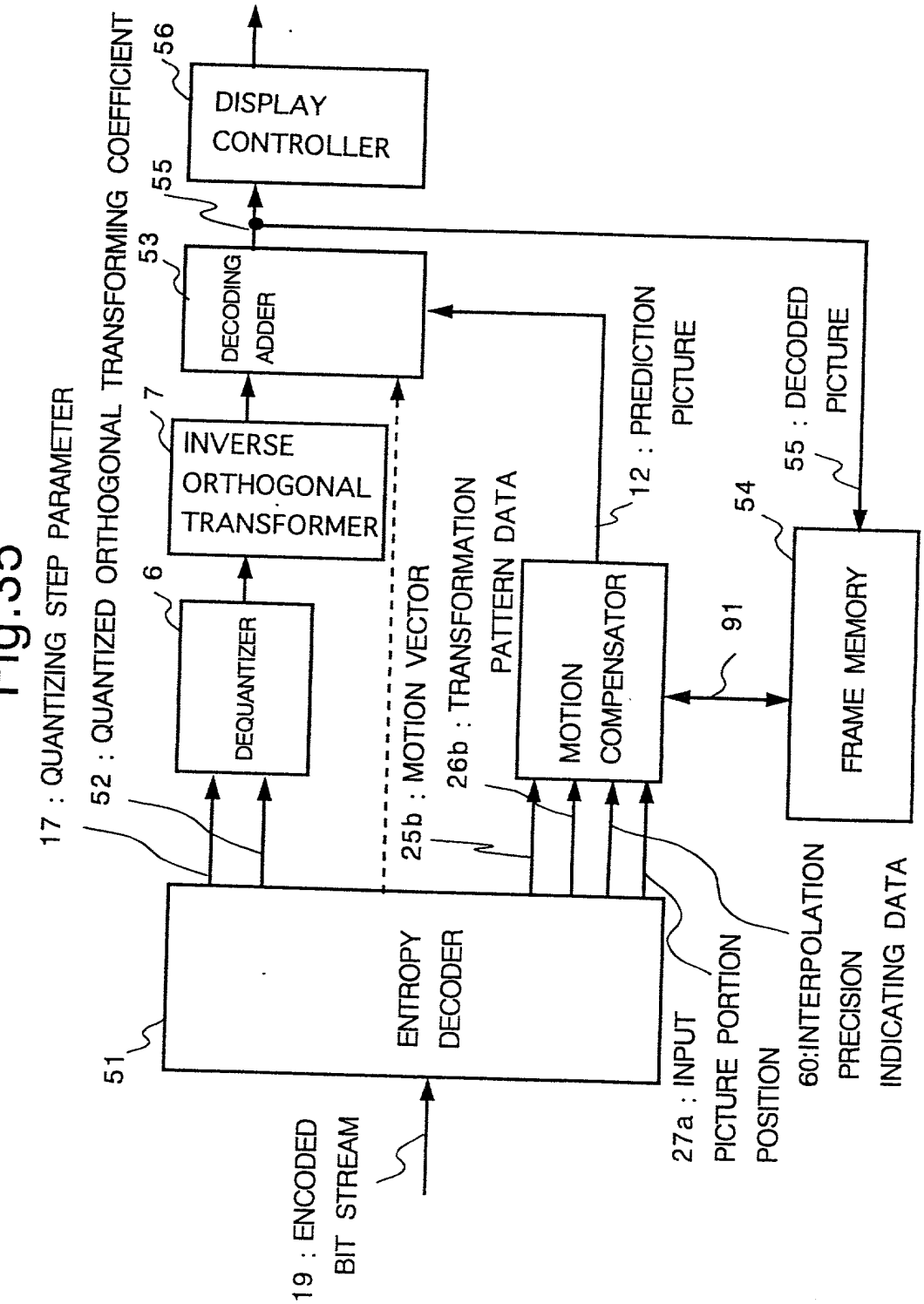

17 : QUANTIZING STEP PARAMETER
52 : QUANTIZED ORTHOGONAL TRANSFORMING COEFFICIENT
25b : MOTION VECTOR
26b : TRANSFORMATION PATTERN DATA
12 : PREDICTION PICTURE
55 : DECODED PICTURE
91
60:INTERPOLATION PRECISION INDICATING DATA
27a : INPUT PICTURE PORTION POSITION
19 : ENCODED BIT STREAM
51
ENTROPY DECODER
6
DEQUANTIZER
7
INVERSE ORTHOGONAL TRANSFORMER
53
DECODING ADDER
55
56
DISPLAY CONTROLLER
MOTION COMPENSATOR
54
FRAME MEMORY

Fig.36

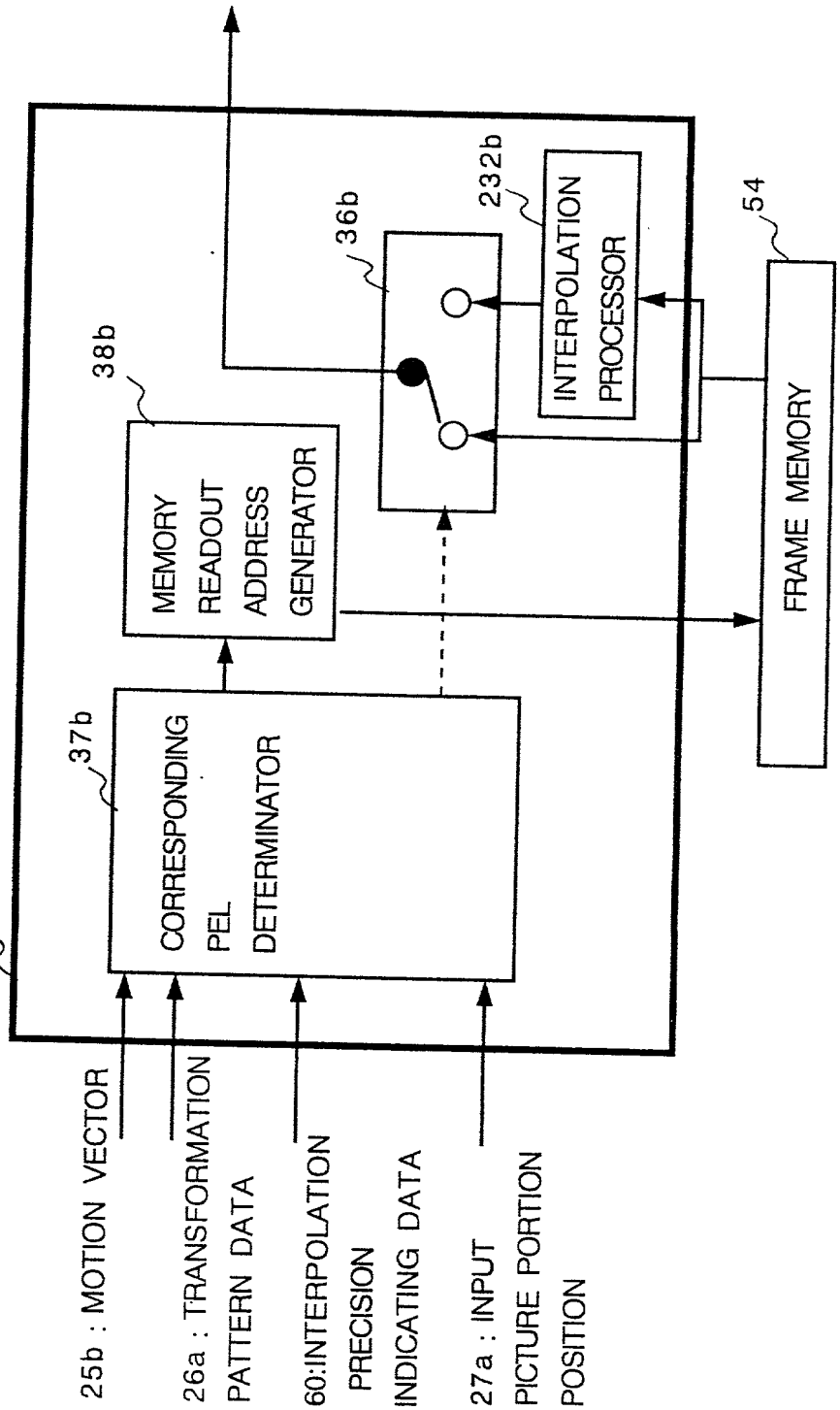

90 : MOTION COMPENSATOR
37b
CORRESPONDING PEL DETERMINATOR
38b
MEMORY READOUT ADDRESS GENERATOR
36b
232b
INTERPOLATION PROCESSOR
54
FRAME MEMORY
25b : MOTION VECTOR
26a : TRANSFORMATION PATTERN DATA
60:INTERPOLATION PRECISION INDICATING DATA
27a : INPUT PICTURE PORTION POSITION

Fig.37

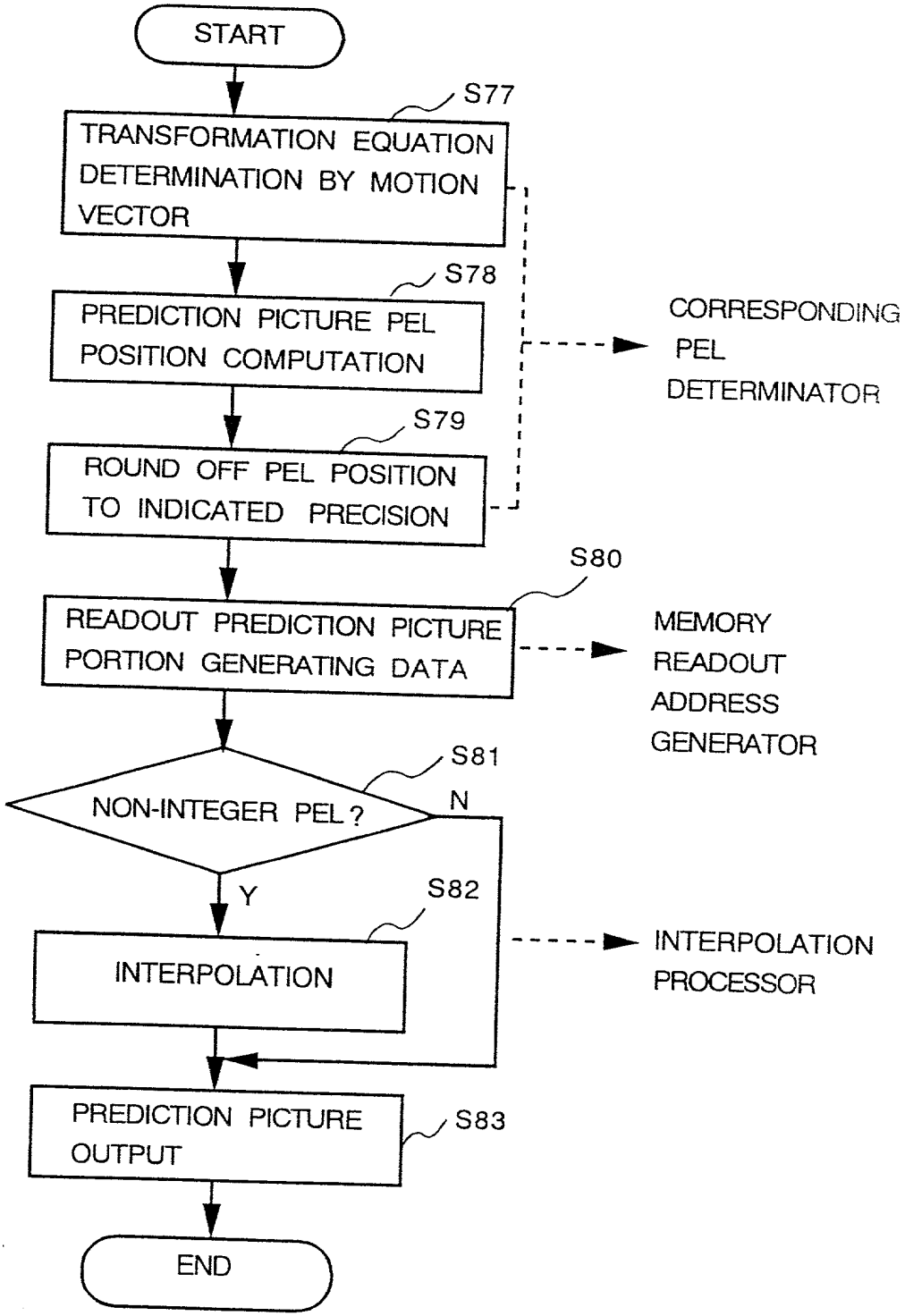
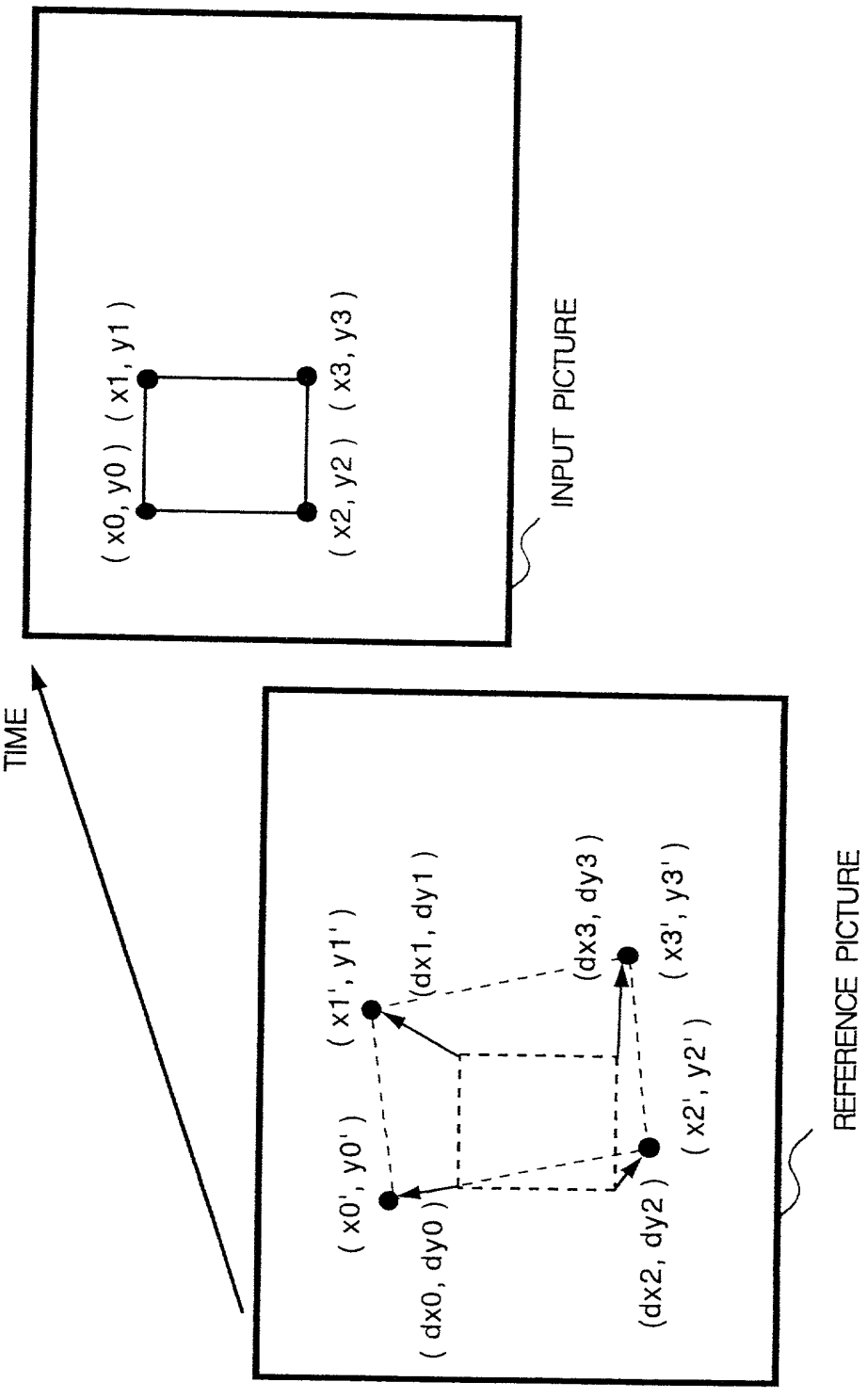
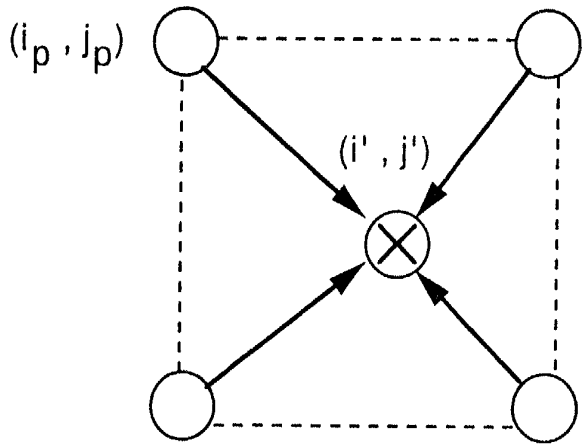

START
S77
TRANSFORMATION EQUATION DETERMINATION BY MOTION VECTOR
S78
PREDICTION PICTURE PEL POSITION COMPUTATION
CORRESPONDING PEL DETERMINATOR
S79
ROUND OFF PEL POSITION TO INDICATED PRECISION
S80
READOUT PREDICTION PICTURE PORTION GENERATING DATA
MEMORY READOUT ADDRESS GENERATOR
S81
NON-INTEGER PEL?
N
Y
S82
INTERPOLATION
INTERPOLATION PROCESSOR
PREDICTION PICTURE OUTPUT
S83
END

Fig.38


TIME
(x0, y0)
(x1, y1)
(x2, y2)
(x3, y3)
INPUT PICTURE
(x0', y0')
(x1', y1')
(x2', y2')
(x3', y3')
(dx0, dy0)
(dx1, dy1)
(dx2, dy2)
(dx3, dy3)
REFERENCE PICTURE

20mUED 9tm98007

Fig.39



$$\hat{I}(i' , j') = W_{x1}W_{y1}I(i_p , j_p) + W_{x2}W_{y1}I(i_{p+1} , j_p) + W_{x2}W_{y1}I(i_{p+1} , j_p+1) + W_{x2}W_{y2}I(i_{p+1} , j_p)$$

$$W_{x2} = i' - j_p$$

$$W_{x1} = 1.0 - W_{x2}$$

$$W_{y2} = j' - j_p$$

$$W_{y1} = 1.0 - W_{y2}$$

Fig.40

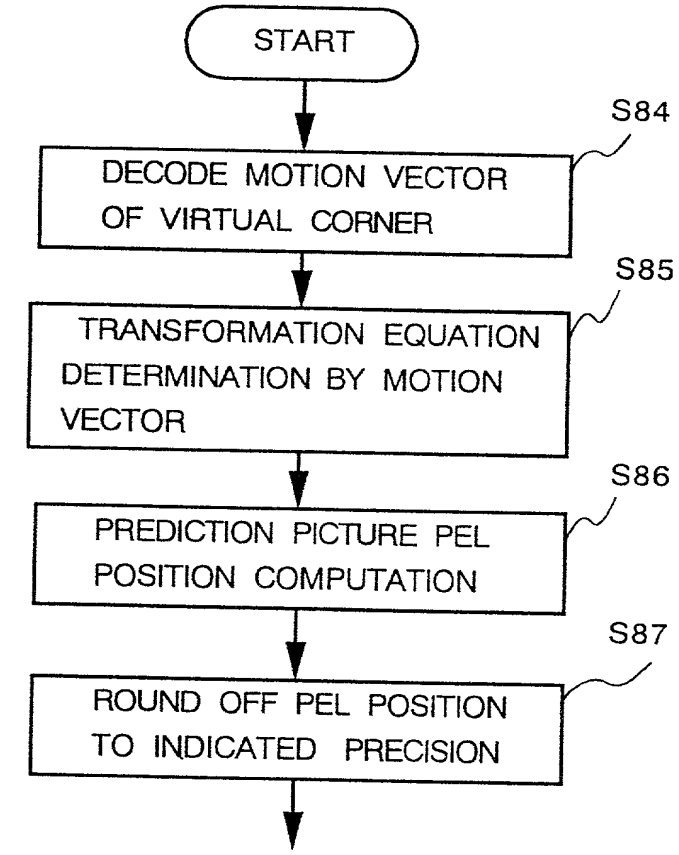

START
S84
DECODE MOTION VECTOR OF VIRTUAL CORNER
S85
TRANSFORMATION EQUATION DETERMINATION BY MOTION VECTOR
S86
PREDICTION PICTURE PEL POSITION COMPUTATION
S87
ROUND OFF PEL POSITION TO INDICATED PRECISION
TO PREDICTION PICTURE GENERATION

Fig.41

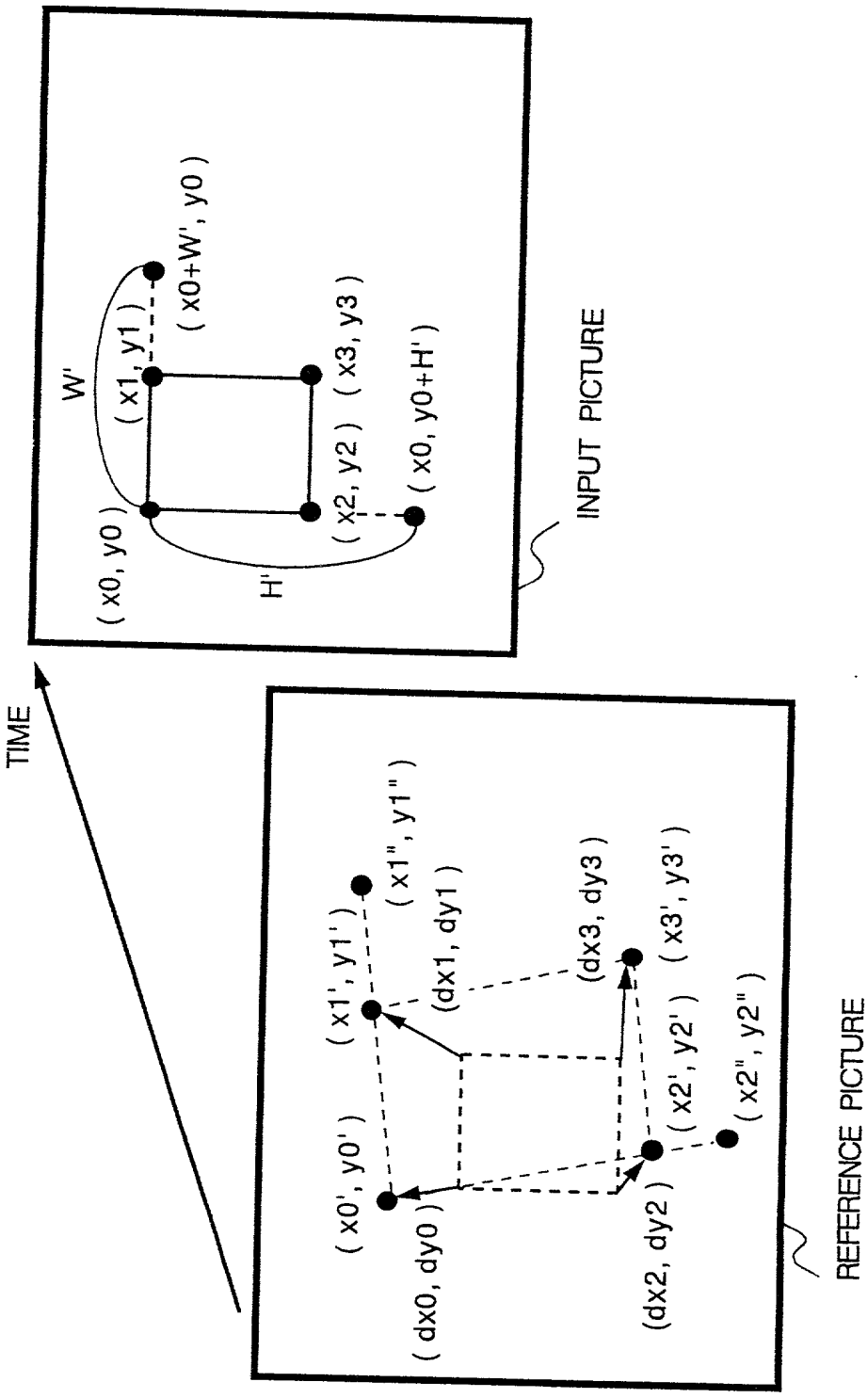

TIME
INPUT PICTURE
REFERENCE PICTURE
W'
H'
(x0, y0)
(x1, y1)
(x0+W', y0)
(x2, y2)
(x3, y3)
(x0, y0+H')
(x0', y0')
(x1', y1')
(x1", y1")
(dx0, dy0)
(dx1, dy1)
(dx2, dy2)
(dx3, dy3)
(x2', y2')
(x3', y3')
(x2", y2")

Fig.42

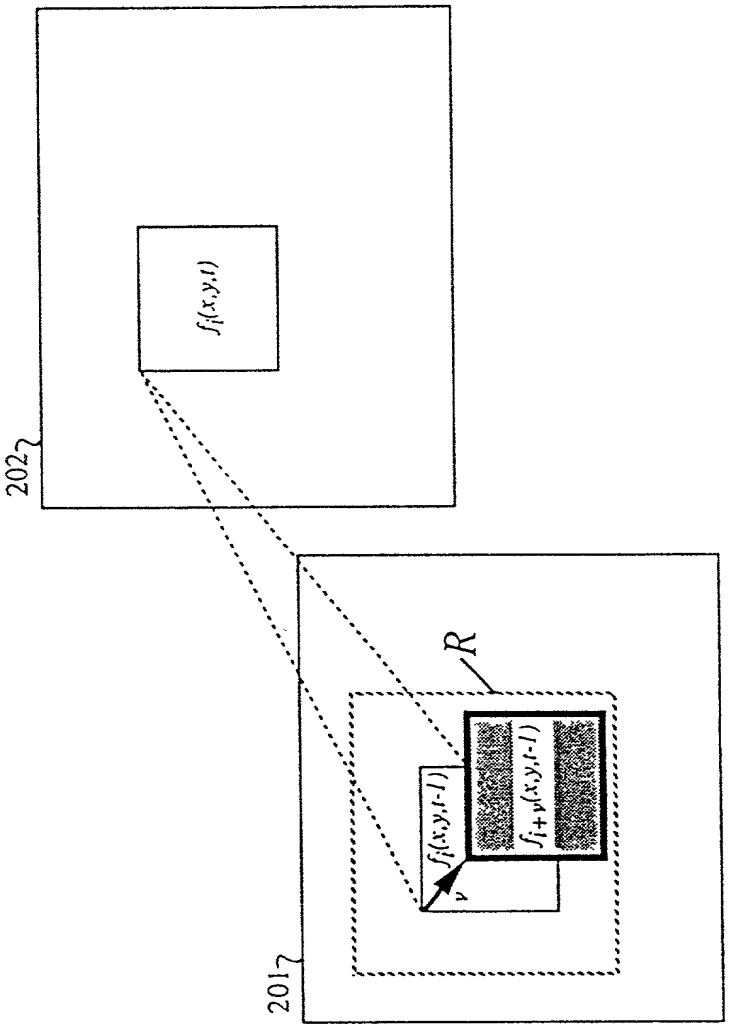


Fig.43

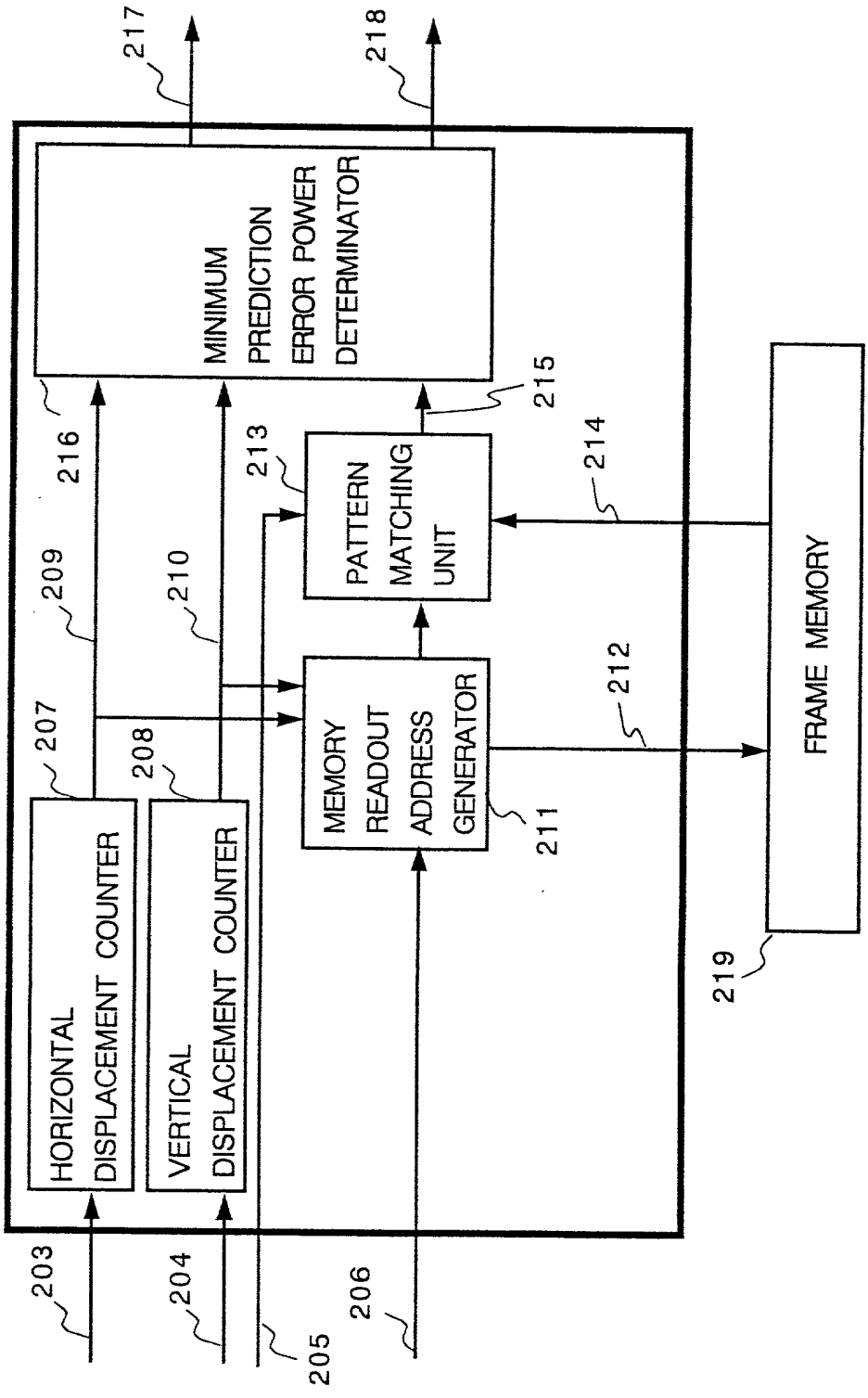

HORIZONTAL DISPLACEMENT COUNTER
VERTICAL DISPLACEMENT COUNTER
MEMORY READOUT ADDRESS GENERATOR
PATTERN MATCHING UNIT
MINIMUM PREDICTION ERROR POWER DETERMINATOR
FRAME MEMORY
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219

Fig.44

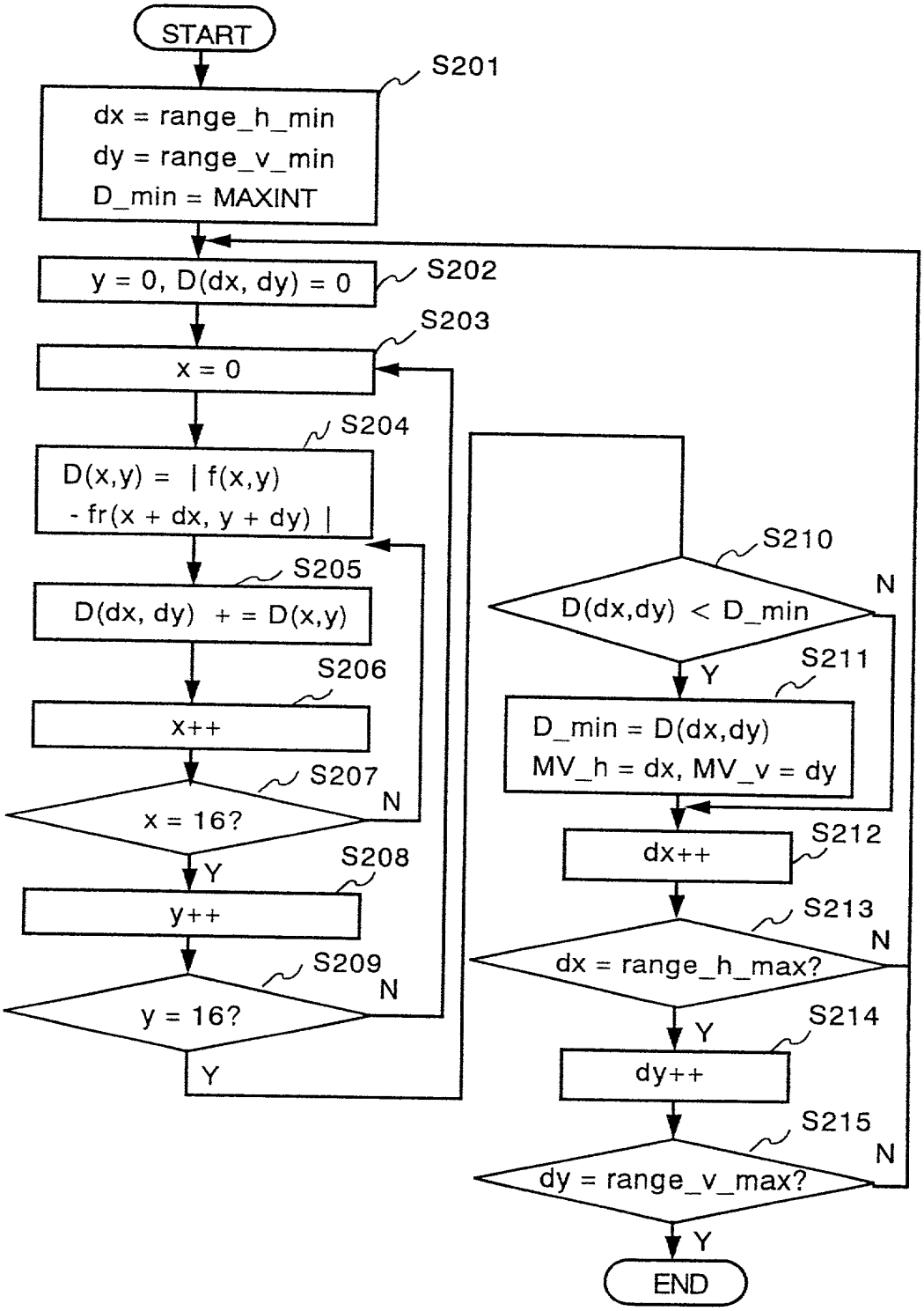

START
S201
dx = range_h_min
dy = range_v_min
D_min = MAXINT
y = 0, D(dx, dy) = 0
S202
S203
x = 0
S204
D(x,y) = | f(x,y)
- fr(x + dx, y + dy) |
S205
D(dx, dy) + = D(x,y)
S206
x++
S207
x = 16?
N
Y
S208
y++
S209
y = 16?
N
Y
S210
D(dx,dy) < D_min
N
Y
S211
D_min = D(dx,dy)
MV_h = dx, MV_v = dy
S212
dx++
S213
dx = range_h_max?
N
Y
S214
dy++
S215
dy = range_v_max?
N
Y
END

Fig.45

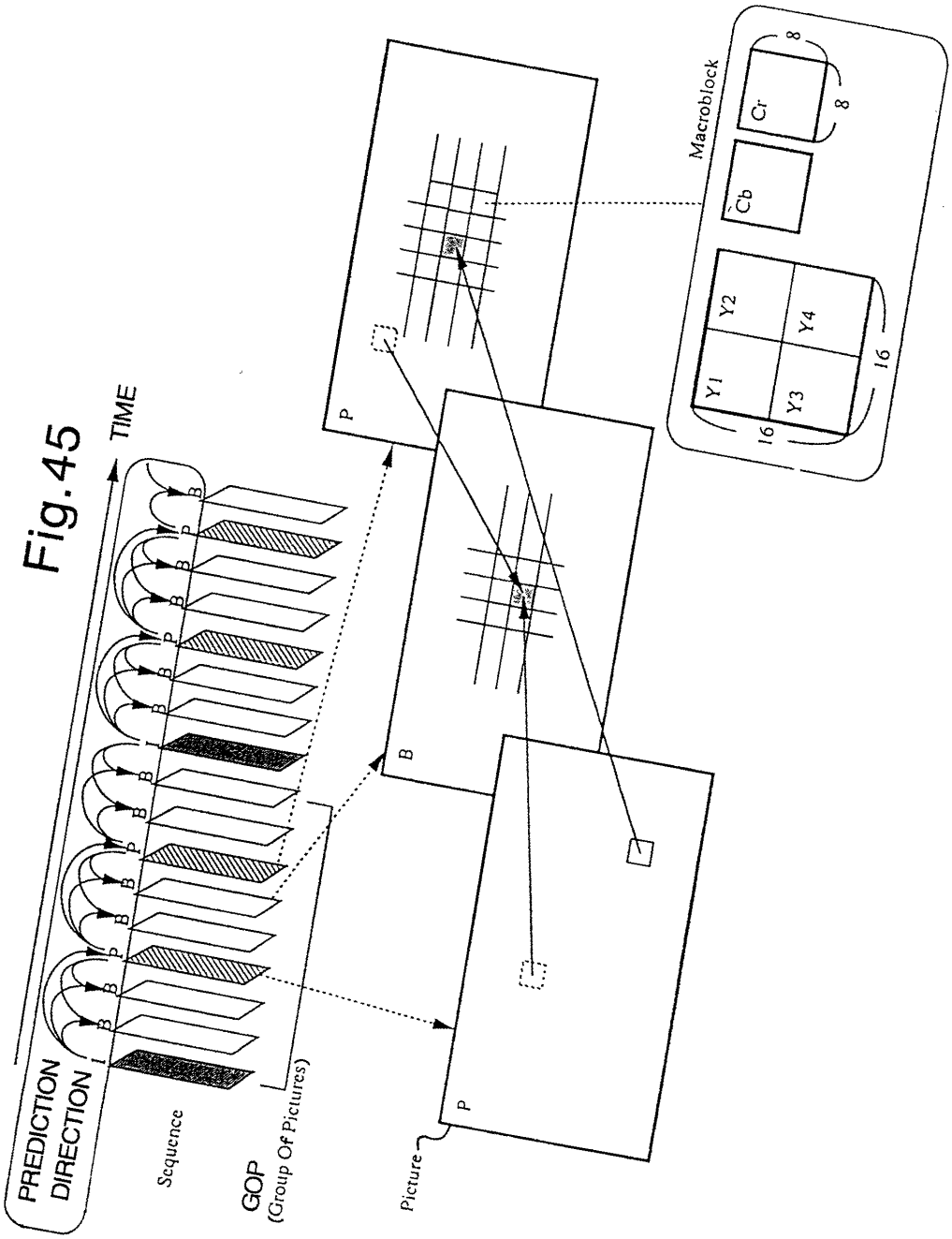

TIME
PREDICTION DIRECTION
Sequence
GOP (Group Of Pictures)
Picture
I
P
B
P
Macroblock
Y1
Y2
Y3
Y4
16
16
Cb
Cr
8
8

Fig.46

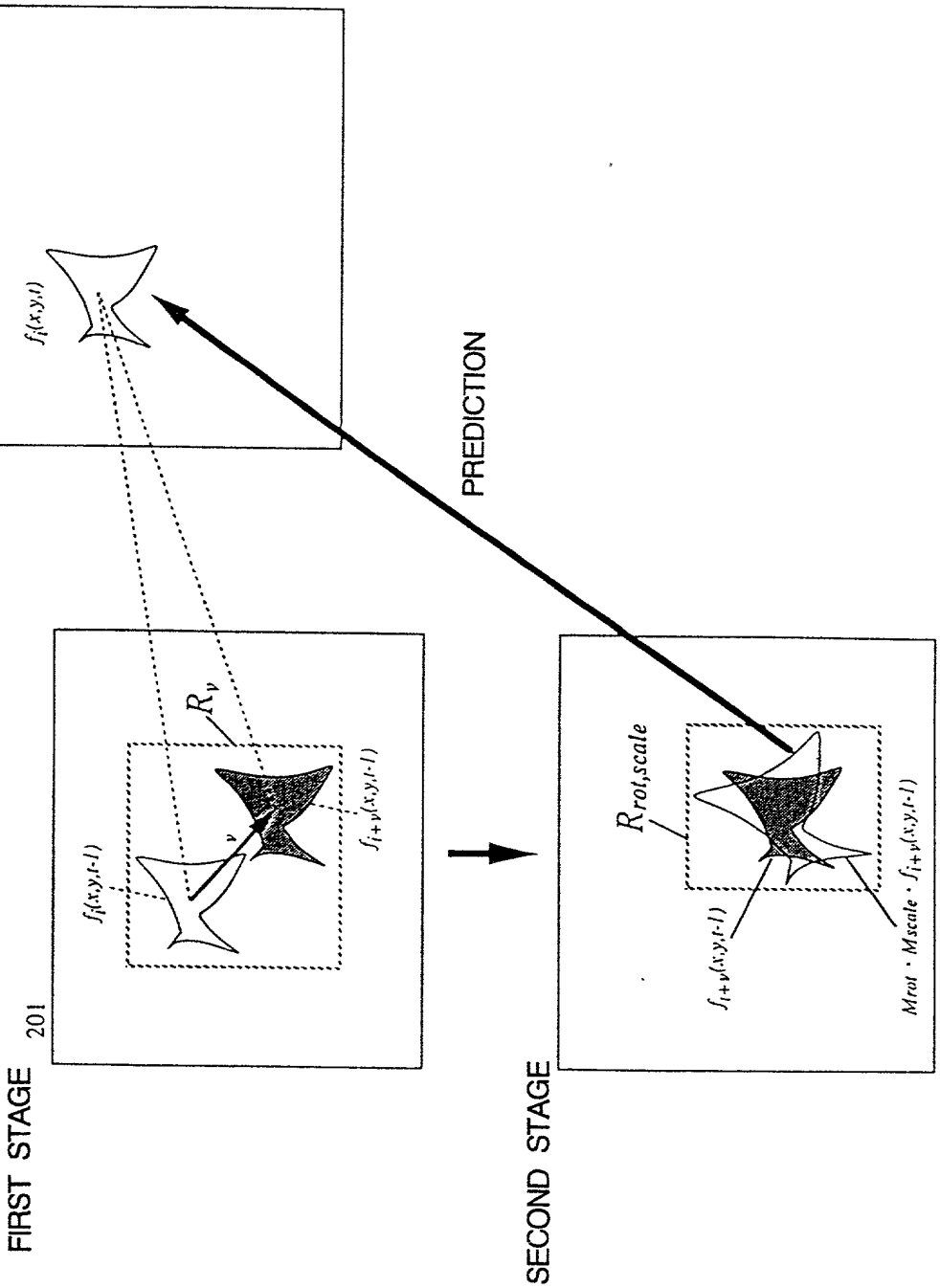

FIRST STAGE
201
202
$f_t(x,y,t)$
$f_t(x,y,t-1)$
R_v
v
$f_{t+\mu}(x,y,t-1)$
PREDICTION
SECOND STAGE
$R_{rot,scale}$
$f_{t+\mu}(x,y,t-1)$
$Mrot \cdot Mscale \cdot f_{t+\mu}(x,y,t-1)$

Fig.47

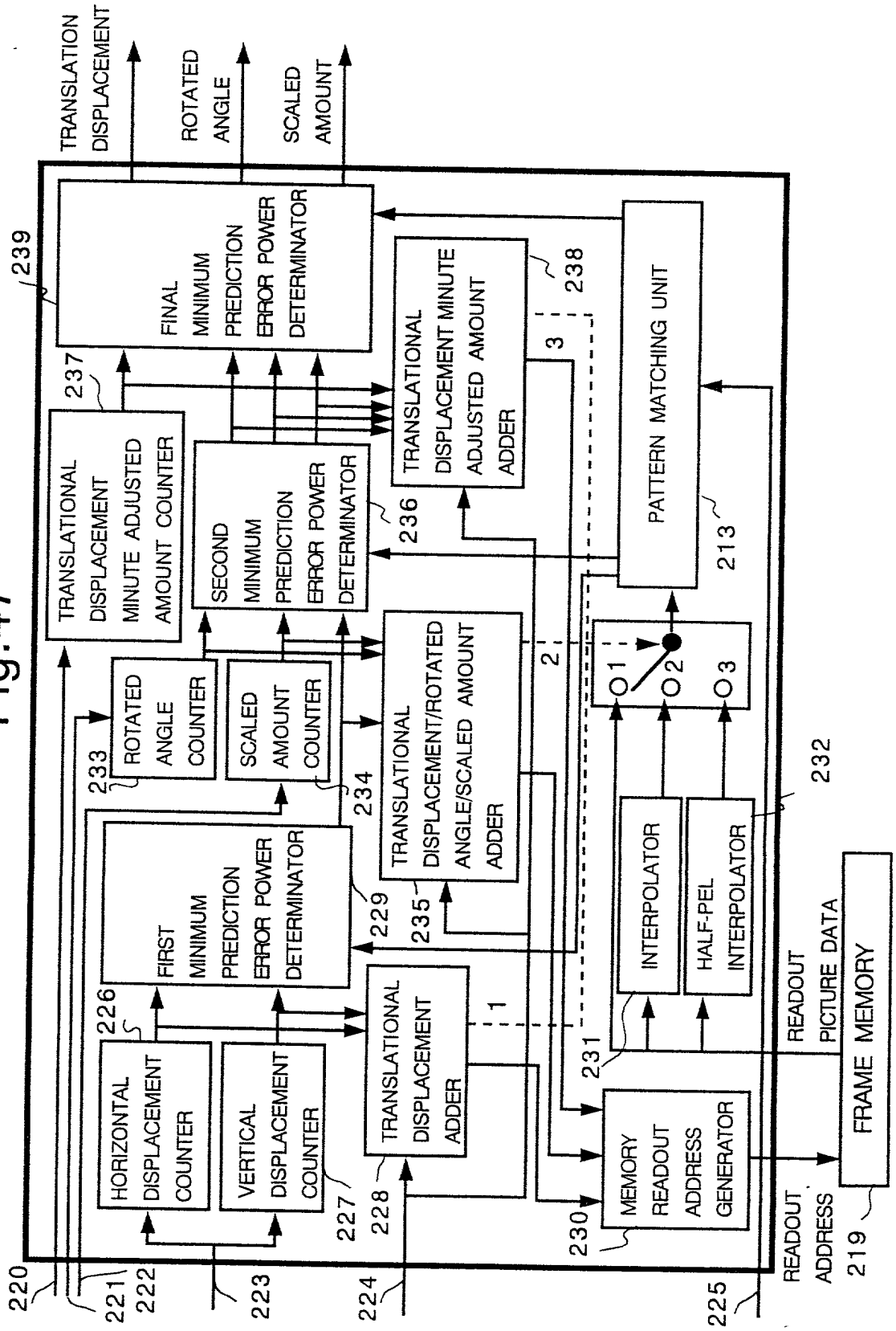

TRANSLATION DISPLACEMENT
ROTATED ANGLE
SCALED AMOUNT
239
FINAL MINIMUM PREDICTION ERROR POWER DETERMINATOR
237
TRANSLATIONAL DISPLACEMENT MINUTE ADJUSTED AMOUNT COUNTER
238
TRANSLATIONAL DISPLACEMENT MINUTE ADJUSTED AMOUNT ADDER
236
SECOND MINIMUM PREDICTION ERROR POWER DETERMINATOR
PATTERN MATCHING UNIT
213
233
ROTATED ANGLE COUNTER
234
SCALED AMOUNT COUNTER
TRANSLATIONAL DISPLACEMENT/ROTATED ANGLE/SCALED AMOUNT ADDER
229
FIRST MINIMUM PREDICTION ERROR POWER DETERMINATOR
235
226
HORIZONTAL DISPLACEMENT COUNTER
VERTICAL DISPLACEMENT COUNTER
227
TRANSLATIONAL DISPLACEMENT ADDER
228
INTERPOLATOR
HALF-PEL INTERPOLATOR
232
231
MEMORY READOUT ADDRESS GENERATOR
230
READOUT PICTURE DATA
FRAME MEMORY
READOUT ADDRESS
219
220
221
222
223
224
225

Fig.48

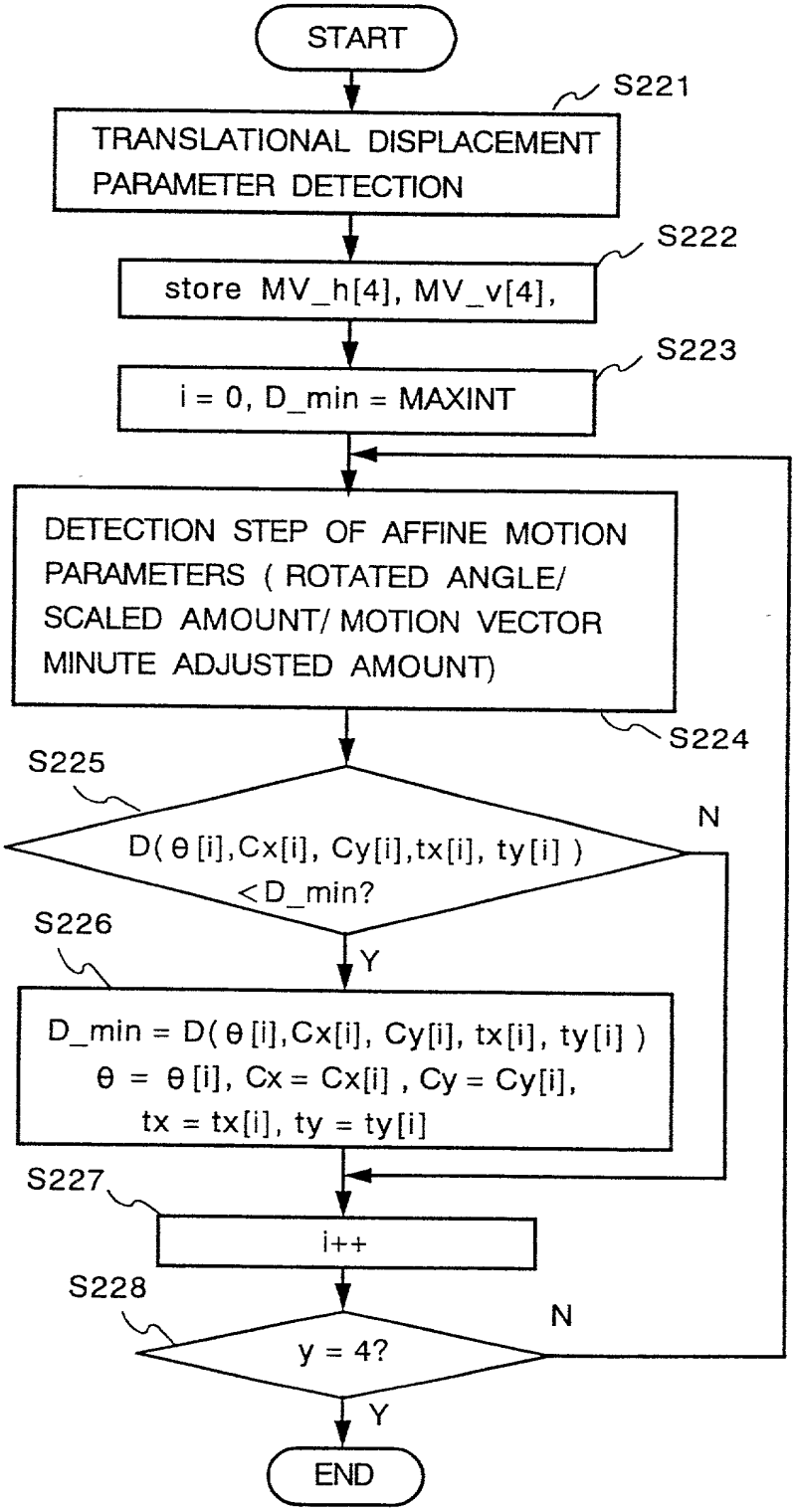

START
S221
TRANSLATIONAL DISPLACEMENT PARAMETER DETECTION
S222
store MV_h[4], MV_v[4],
S223
i = 0, D_min = MAXINT
DETECTION STEP OF AFFINE MOTION PARAMETERS (ROTATED ANGLE/ SCALED AMOUNT/ MOTION VECTOR MINUTE ADJUSTED AMOUNT)
S224
S225
D(θ [i],Cx[i], Cy[i],tx[i], ty[i]) < D_min?
N
Y
S226
D_min = D(θ [i],Cx[i], Cy[i], tx[i], ty[i])
θ = θ [i], Cx = Cx[i] , Cy = Cy[i],
tx = tx[i], ty = ty[i]
S227
i++
S228
y = 4?
N
Y
END

Fig.49

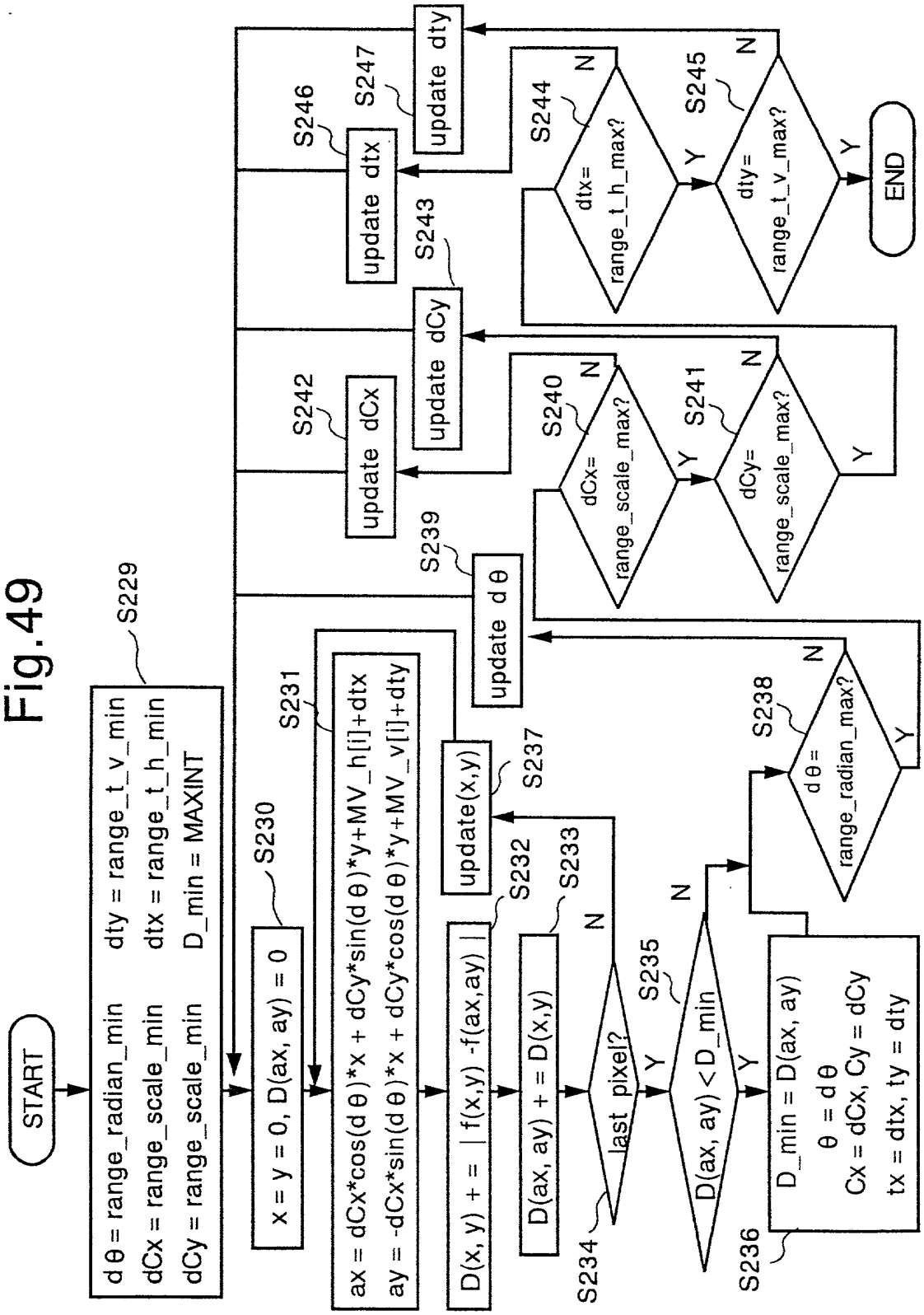

START
S229
d θ = range_radian_min
dCx = range_scale_min
dCy = range_scale_min
dty = range_t_v_min
dtx = range_t_h_min
D_min = MAXINT
S230
x = y = 0, D(ax, ay) = 0
S231
ax = dCx*cos(d θ)*x + dCy*sin(d θ)*y+MV_h[i]+dtx
ay = -dCx*sin(d θ)*x + dCy*cos(d θ)*y+MV_v[i]+dty
S232
D(x, y) + = | f(x,y) -f(ax,ay) |
S233
D(ax, ay) + = D(x,y)
S234
last pixel?
S235
D(ax, ay) <D_min
S236
D_min = D(ax, ay)
θ = d θ
Cx = dCx, Cy = dCy
tx = dtx, ty = dty
S237
update(x,y)
S238
d θ =
range_radian_max?
S239
update d θ
S240
dCx=
range_scale_max?
S241
dCy=
range_scale_max?
S242
update dCx
S243
update dCy
S244
dtx=
range_t_h_max?
S245
dty=
range_t_v_max?
S246
update dtx
S247
update dty
END
Y
N

Fig.50

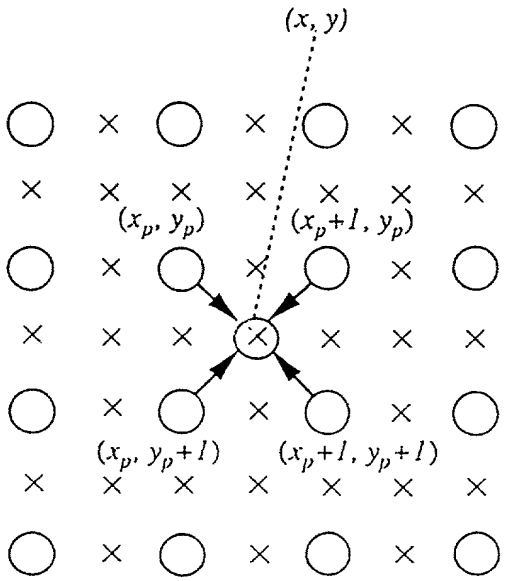
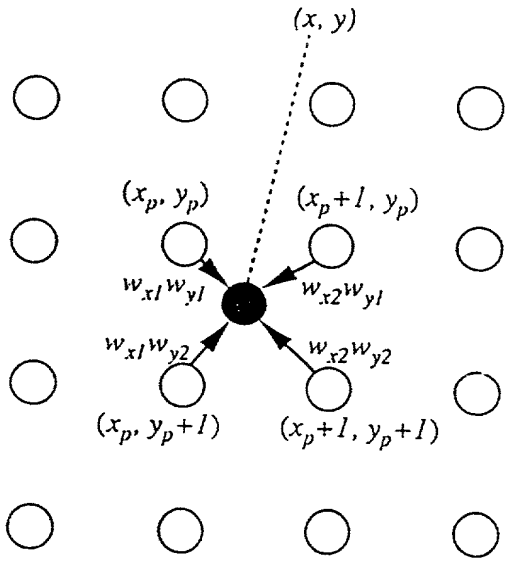



Fig.51